Allen-Bradley

# Speicherprogrammierbare Steuerungen MicroLogix™ 1200 und MicroLogix 1500

**Bulletins 1762 und 1764**

**Befehlssatz Referenzhandbuch**

Rockwell Automation

## Wichtige Hinweise für Benutzer

Aufgrund der vielfältigen Einsatzmöglichkeiten der in dieser Publikation beschriebenen Produkte müssen die für die Anwendung und den Einsatz dieses Geräts verantwortlichen Personen sicher- stellen, dass jede Anwendung bzw. jeder Einsatz alle Leistungs- und Sicherheitsanforderungen, einschließlich sämtlicher anwendbarer Gesetze, Vorschriften, Bestimmungen und Normen erfüllt. Rockwell Automation ist in keinem Fall verantwortlich oder haftbar für indirekte Schäden oder Folgeschäden, die durch den Einsatz oder die Anwendung dieser Produkte entstehen.

Die Abbildungen, Diagramme, Beispielprogramme und Aufbaubeispiele in dieser Publikation dienen ausschließlich zur Veranschaulichung. Aufgrund der unterschiedlichen Anforderungen der jeweiligen Applikation kann Rockwell Automation keine Verantwortung oder Haftung (einschließlich Haftung für geistiges Eigentum) für den tatsächlichen Einsatz auf der Grundlage dieser Beispiele übernehmen.

In der Allen-Bradley-Publikation SGI-1.1 *Safety Guidelines for the Application, Installation and Maintenance of Solid-State Control* (erhältlich bei Ihrem Rockwell Automation-Vertriebsbüro) werden einige wichtige Unterschiede zwischen elektronischen und elektromechanischen Geräten erläutert. Diese müssen bei der Verwendung der in diesem Handbuch beschriebenen Produkte berücksichtigt werden.

Die Vervielfältigung des Inhalts dieser urheberrechtlich geschützten Publikation, in seiner Gesamtheit oder in Teilen, ohne die schriftliche Einwilligung von Rockwell Automation ist verboten.

Besondere Hinweise in dieser Publikation sollen den Anwender auf mögliche Gefahrenzustände aufmerksam machen. Die folgenden Hinweise und die daneben aufgeführten Aussagen helfen Ihnen dabei, potenzielle Gefahrenzustände zu erkennen, die potenzielle Gefahr zu vermeiden und die Folgen der potenziellen Gefahr zu erkennen:

**WARNUNG**

Dieser Hinweis macht Sie auf Vorgehensweisen und Zustände aufmerksam, die in Gefahrenumgebungen eine Explosion verursachen können. Diese können zu Verletzungen oder Tod, Sachschäden oder wirtschaftlichen Verlusten führen.

**ACHTUNG**

Dieser Hinweis macht Sie auf Vorgehensweisen und Zustände aufmerksam, die zu Verletzungen oder Tod, Sachschäden oder wirtschaftlichen Verlusten führen können.

**WICHTIG**

Kennzeichnet Informationen, die unabdingbar sind für die erfolgreiche Verwendung des Produkts.

Im Folgenden sind die Änderungen in diesem Handbuch seit der letzten Drucklegung als Publikation 1762-RM001C-DE-P, November 2000, zusammengefasst.

## Überblick zur Firmwarerevision

Die Steuerungen werden über Firmwareaktualisierungen um Leistungsmerkmale erweitert. Anhand der nachfolgenden Liste können Sie überprüfen, ob die Firmware Ihrer Steuerung sich auf dem erforder- lichen Stand befindet. Firmwareaktualisierungen sind nur erforderlich, wenn Sie die neuen Leistungsmerkmale nutzen möchten. *Siehe „Firmware-Upgrades" auf Seite iii.*

**MicroLogix 1200**

| Bestellnummer | Serien-buchstabe | Revisions-buchstabe | Firmware-Release-Nr. | Release-Datum | Verbesserung |
|---|---|---|---|---|---|
| 1762-L24AWA<br>1762-L24BWA<br>1762-L40AWA<br>1762-L40BWA | A | A | FRN1 | März 2000 | Produkt-Release. |
| | A | B | FRN2 | Mai 2000 | Die Einstellpotentiometer der Steuerungen funktionierten umgekehrt zur Kontaktplanlogik. Korrigiert. |
| | B | A | FRN3 | November 2000 | MicroLogix 1200-Steuerungen bieten jetzt folgende Vorteile:<br>• Voll-ASCII (Lesen/Schreiben)<br>• PTO-gesteuerter Halt<br>• PWM-Rampen<br>• RTC- und Zeichenketten-Nachrichtenfunktion<br>• Statischer Datenfileschutz<br>• Drucktaster-Bit zum Zurücksetzen der Kommunikation |
| 1762-L24BXB<br>1762-L40BXB | B | A | FRN3 | November 2000 | Produkt-Release. Unterstützt alle oben aufgeführten Leistungsmerkmale für die 1762-L24xWA- und 1762-L40xWA- Steuerungen. |
| 1762-L24AWA<br>1762-L24BWA<br>1762-L24BXB<br>1762-L40AWA<br>1762-L40BWA<br>1762-L40BXB | C | A | FRN4(1) | Juni 2001 | MicroLogix 1200-Steuerungen bieten jetzt folgende Vorteile:<br>• Fließkomma (F)-Datenfile für: Vergleichsbefehle (EQU, GEQ, GRT, LEQ, LES, LIM, NEQ); mathematische Befehle (ABS, ADD, CLR, DIV, MUL, NEG, SQR, SUB); den Verschiebebefehl (MOV); Filebefehle (CPW, FLL) und den Nachrichtenbefehl (MSG)<br>• Programmierbaren Endschalter-File (PLS) für den Hochgeschwindigkeits-Zähler (HSC)<br>• RTA – Einstellen der Echtzeituhr<br>• GCD – Gray-Code<br>• CPW – Wort kopieren<br>• ABS – Absolutwert |
| | C | B | FRN5(2) | März 2002 | Interne Firmwarerevision; keine Funktionalitätsänderung für Benutzer. |
| | C | C | FRN6(2) | Septem-ber 2002 | MicroLogix 1200-Steuerungen bieten jetzt folgende Vorteile:<br>• Der Fließkomma (F)-Datenfile kann nun zur Skalierung mit Parametern (SCP) verwendet werden<br>• Verbesserungen der Modbus-Speicherbelegung |

Auf der MicroLogix-Website (www.ab.com/micrologix) finden Sie ältere Versionen und Aktualisierungen des Flashs der Betriebssystem-Firmware für MicroLogix 1200-Steuerungen. Jede Steuerung kann mithilfe dieser Werkzeuge auf die neueste Version aktualisiert werden. Informationen zur Herunterstufung finden Sie im folgenden Abschnitt.

(1) RSLogix 500-Programmiersoftware, Version 4.5 – MicroLogix 1200-Steuerungen der Serie C, Revision A mit FRN4-Firmware können mithilfe des über die Website von MicroLogix erhältlichen Werkzeugs ControlFlash FRN3 heruntergestuft werden, sodass sie mit dieser Softwareversion kompatibel sind. Die Steuerung kann später mithilfe des FRN5 (ersetzt das FRN4-ControlFlash-Upgrade und besitzt gleichwertige Funktionen) oder eines höheren ControlFlash-Werkzeugs aktualisiert werden.

(2) RSLogix 500-Programmiersoftware, Version 4.5 – MicroLogix 1200-Steuerungen der Serie C, Revision B mit FRN5 oder höherer Firmware können mithilfe des über die Website von MicroLogix erhältlichen Werkzeugs ControlFlash FRN3.1 heruntergestuft werden, sodass sie mit dieser Softwareversion kompatibel sind. Die Steuerung kann später mithilfe des FRN5 (ersetzt das FRN4-ControlFlash-Upgrade und besitzt gleichwertige Funktionen) oder eines höheren ControlFlash-Werkzeugs aktualisiert werden.

**MicroLogix 1500**

| Bestell-nummer | Serien-buchstabe | Revisions-buchstabe | Firmware-Release-Nr. | Release-Datum | Verbesserung |
|---|---|---|---|---|---|
| 1764-LSP | A | B | FRN2 | Februar 1999 | Produkt-Release. |
| 1764-LSP | A | C | FRN3 | Oktober 1999 | MicroLogix 1500-Steuerungen mit 1764-LSP-Prozessor können jetzt zusammen mit Erweiterungskabeln und Netzteilen für Compact I/O (Bulletin 1769) verwendet werden. |
| 1764-LSP | B | A | FRN4 | April 2000 | MicroLogix 1500-Steuerungen mit 1764-LSP-Prozessor können jetzt Folgendes verwenden:<br>• Zeichenketten-Datenfile-Typ<br>• Unterstützung des ASCII-Befehlssatzes<br>• Modbus-RTU-Slave-Protokoll<br>• Rampenfunktion bei Verwendung von PWM-Ausgängen<br>• Statischer Datenfileschutz<br>• RTC-Nachrichtenfunktion |
| 1764-LRP | B | A | FRN4 | April 2000 | Produkt-Release. MicroLogix 1500-Steuerungen mit 1764-LRP-Prozessor verfügen über alle Leistungsmerkmale des 1764-LSP. Sie bieten zusätzlich:<br>• Einen zweiten Kommunikationsanschluss (RS-232 isoliert)<br>• Funktionen zur Datenprotokollierung |
| 1764-LSP<br>1764-LRP | B | B | FRN5 | Oktober 2000 | Für die 1764-LSP- und LRP-Prozessoren:<br>• Bei Einsatz der PTO-Funktion kann die Steuerung jetzt einen kontrollierten Halt durchführen, wenn PTO-Ausgänge verwendet werden. Die Verzögerungsphase des PTO kann über die Kontaktplanlogik früher initiiert werden.<br>• Erweiterte Funktion für Programmvergleichs-Bits im Speichermodul. |
| 1764-LSP<br>1764-LRP | C | A | FRN6 | September 2001 | MicroLogix 1500-Steuerungen bieten jetzt folgende Vorteile:<br>• Fließkomma (F)-Datenfile für: Vergleichsbefehle (EQU, GEQ, GRT, LEQ, LES, LIM, NEQ); mathematische Befehle (ABS, ADD, CLR, DIV, JUL, NEG, SQR, SUB); den Verschiebebefehl (MOV); Filebefehle (CPW, FLL) und den Nachrichtenbefehl (MSG)<br>• Programmierbaren Endschalter-File (PLS) für den Hochgeschwindigkeits-Zähler (HSC)<br>• RTA – Einstellen der Echtzeituhr<br>• GCD – Gray-Code<br>• CPW – Wort kopieren<br>• ABS – Absolutwert<br>• RCP – Rezept<br>• MSG – Nachricht über DeviceNet *(nur 1764-LRP)* |
| 1764-LSP<br>1764-LRP | C | B | FRN7 | September 2002 | MicroLogix 1500-Steuerungen bieten jetzt folgende Vorteile:<br>• Der Fließkomma (F)-Datenfile kann nun zur Skalierung mit Parametern (SCP) verwendet werden<br>• Verbesserungen der Modbus-Speicherbelegung |

## Firmware-Upgrades

Mithilfe eines Firmware-Upgrades werden der Steuerung erweiterte Funktionen hinzugefügt. *Firmware-Upgrades sind nur erforderlich, wenn Sie die neuen Leistungsmerkmale nutzen möchten.* Um die neuen Funktionen nutzen zu können, vergewissern Sie sich, dass sich die Firmware der Steuerung auf dem folgenden Stand befindet:

| Programmierbare Steuerung | Firmwarerevision | Bestellnummern |
|---|---|---|
| MicroLogix 1200 | Serie C, Revision C, FRN6 | Steuerungen 1762-L24AWA, -L24BWA, -L24BXB, -L40AWA, -L40BWA und -L40BXB |
| MicroLogix 1500 | Serie C, Revision B, FRN7 | Prozessoren 1764-LSP und -LRP |

Die Firmware für eine MicroLogix-Steuerung können Sie über die MicroLogix-Website unter **http://www.ab.com/micrologix** aktualisieren.

Um die neuen Funktionen nutzen zu können, muss die RSLogix 500-Programmiersoftware, Version 5.50 oder höher, installiert sein.

## Neue Informationen

In der folgenden Tabelle sind die Seiten aufgeführt, auf denen neue Informationen enthalten sind.

| Neue Informationen | Siehe Seite |
|---|---|
| Überarbeiteter Abschnitt zu *Rockwell Automation-Support*. | Seite 1 |
| Zusätzliche Tabelle 1.1, Gültige Eingangs-/Ausgangs-Datenwortformate, Bereich 0 bis 10 V DC und 4 bis 20 mA im Analogbereich. | 1-5 |
| Zusätzlicher Eingangsdatenfile für 1762-IR4-RTD-/Widerstandsmodul. | 1-7 |
| Zusätzlicher Eingangsdatenfile für 1762-IT4-Thermoelementmodul. | 1-8 |
| Geändert: 8 E/A zu 16 E/A. | 1-9, 1-21, 3-19 |
| Zusätzliche Eingangs- und Ausgangsbilder für 1769-OA16- und 1769-OW16-Module. | 1-12 |
| Zusätzliche Eingangs- und Ausgangsbilder für 1769-IF4XOF2. | 1-14 |
| Zusätzlicher Eingangsdatenfile für 1769-IR6. | 1-16 |
| Korrigierte Bit-Definition, O1, in der Tabelle des Eingangsdatenfiles. | 1-13 |
| Zusätzliche Ausgangsanordnung für das 1769-HSC-Hochgeschwindigkeits-Zählermodul. | 1-18 |
| Zusätzliche Datenorganisation für das 1769-SDN-DeviceNet-Scanner-Modul. | 1-20 |
| Geändertes Format des Abschnitts *Filestruktur* (von einer Abbildung in eine Tabelle) und Hinzufügen von Fließkomma (F), Rezept, programmierbarem Endschalter (PLS) sowie Datenprotokollierfiles und Fußnote 3. | 2-2 |
| Zusätzliche Informationen zu den neuen Datenfiles für Fließkomma (F) und Programmierbarer Endschalter (PLS) . | 2-7, 2-8 , 2-10 |
| Zusätzlicher Hinweis zu den Eingangs- und Ausgangsdatenelementen, von denen jedes 3 Wörter verwendet. | 2-3 |
| Aktualisierte Speicherwerte. | 2-5 |
| Zusätzlicher Abschnitt zu der Überprüfung des Steuerung-Speicherverbrauchs. | 2-6 |
| Neustrukturierter Abschnitt zur Echtzeituhr und zusätzliche Anleitung zum Einstellen der Echtzeituhr (RTA). | 3-3, 3-5 |

| Neue Informationen | Siehe Seite |
|---|---|
| Zusätzlicher Hinweis, dass die 1764-DAT-Betriebsbefehle sich im Benutzerhandbuch für MicroLogix 1500 (Publikation 1764-UM001) befinden. | 3-10 |
| MicroLogix 1200 wurde der Fußnote für Tabelle 3.10 hinzugefügt. | 3-14 |
| Aktualisiertes Kapitel *Programmierbefehle – Übersicht*, um die neuen Befehle der Liste hinzuzufügen. | 4-1 |
| Zusätzliche neue Dateitypen für die Tabellen *Gültige Adressierungsmodi und Filetypen*. | 4-2 und im Rest des Handbuchs |
| Zusätzliche neue Fileinformationen zum programmierbaren Endschalter (PLS) im Kapitel zum Hochgeschwindigkeitszähler. | 5-1, 5-28 |
| Überarbeitete Beschreibung für *Ausgang obere Quelle* und *Ausgang untere Quelle* für die Anleitungen zum Hochgeschwindigkeitszähler (HSL). | 5-26 |
| Überarbeitete Erklärung für PTO-Beschleunigungs-/-Verzögerungsimpulse (ADP). | 6-13 |
| Zusätzliche Informationen bezüglich des Fließkomma-Datenfiles. | 10-1, 10-4 |
| Neue Anleitung für den Absolutwert (ABS). | 10-10 |
| Zusätzlicher Hinweis, dass der Fließkomma (F)-Datenfile nun mit der Anleitung für die Skalierung mit Parametern (SCP) verwendet werden kann. | 10-13, 10-14 |
| Zusätzliche Anleitung für den Gray-Code (GCD). | 11-10 |
| Anleitung für Swap (SWP) aus dem Kapitel *Mathematische Befehle* in das Kapitel *Filebefehle* verschoben. | 14-1, 14-19 |
| Neuer Befehl zum Kopieren eines Wortes (CPW). | 14-2 |
| Zusätzliche Informationen bezüglich des Fließkomma-Datenfiles. | 14-5, 14-6 |
| Zusätzlicher Hinweis, dass das Ausführungs-Bit (RN) nicht über den Control (R)-File adressiert werden kann. | 20-6, 20-28 |
| Überarbeiteter Text für die Masken UND und ODER. | 20-21 |
| Neustrukturiertes Kapitel *Kommunikationsbefehle* und neue Nachrichtenversendung über DeviceNet (CIP-generisch). | Kapitel 21 |
| Aktualisierte Ausführungszeit des MSG-Befehls. | 21-5 |
| Aktualisierte Tabellen der Nachrichtenfile-Elemente und zusätzliche Tabelle für Nachrichtenfile-Zielortinformationen, Zielgerät = CIP-generisch. | 21-6, 21-7 |
| Zusätzliche Informationen zum Fließkomma-File. | 21-21, 21-22, 21-24 |
| Neuer Rezept-Befehl (RCP). | 22-1 |
| Zusätzliche Befehle „Absolutwert„ (ABS), „Wort kopieren„ (CPW), „Gray-Code„ (GCD) und „Echtzeituhr einstellen„ (RTA). | Anhang A |
| Zusätzliche Befehle „Absolutwert„ (ABS), „Wort kopieren„ (CPW), „Gray-Code„ (GCD) und „Echtzeituhr einstellen„ (RTA). | Anhang B |
| Zusätzliche Informationen über einen Hardware-Ausfall in Fehlercode 0021. | D-4 |
| Zusätzliche Informationen zur erweiterten Modbus-Speicherbelegung. Die Steuerung unterstützt jetzt bis zu 1536 (vorher 256) Halteregister, die bis zu sechs (vorher 1) Ganzzahlen oder Bits des Datentabellenfiles zugeordnet werden können. | E-9 bis E-13 |
| Neue Befehle (RTA, ABS, GCD, CPW, RCP) unter *Alphabetische Liste der Befehle*. | Innenseite der Rückseite |

Lesen Sie dieses Vorwort, um sich mit dem Rest des Handbuchs vertraut zu machen. Das Vorwort enthält Informationen zu folgenden Themen:

- Zielgruppe dieses Handbuchs
- Zweck dieses Handbuchs
- Literaturhinweise
- Konventionen in diesem Handbuch
- Unterstützung durch Rockwell Automation

## Zielgruppe dieses Handbuchs

Verwenden Sie das vorliegende Handbuch, wenn Sie verantwortlich sind für die Entwicklung, Installation, Programmierung oder Fehler- suche bei Steuerungssystemen, die MicroLogix 1200- oder MicroLogix-1500-Steuerungen verwenden.

Dabei sollten Sie über Grundkenntnisse elektrischer Schaltkreise verfügen und mit der Relaislogik vertraut sein. Ist dies nicht der Fall, sollten Sie vor Verwendung dieses Produkts geeignete Weiterbil- dungskurse belegen.

## Zweck dieses Handbuchs

Das vorliegende Handbuch ist ein Referenzhandbuch für MicroLogix 1200- und MicroLogix 1500-Steuerungen. In diesem Referenzhandbuch werden die Vorgehensweisen zur Installation, Verdrahtung, Programmierung und Fehlerbehebung der Steuerung beschrieben. Im Einzelnen enthält dieses Handbuch:

- eine Übersicht der Filetypen, die von den Steuerungen verwendet werden
- den Befehlssatz für die Steuerungen
- Anwendungsbeispiele für den Befehlssatz.

## In diesem Handbuch verwendendete Konventionen

In diesem Handbuch werden die folgenden Konventionen verwendet:

- Mit Punkten versehene Listen wie diese enthalten Informationen, aber keine Verfahrensweisen.
- Nummerierte Auflistungen enthalten sequenzielle Schritte bzw. hierarchisch angeordnete Informationen.
- *Kursivschrift* wird zur Hervorhebung verwendet.

## Literaturhinweise

Folgende Dokumente enthalten zusätzliche Informationen zu Produkten von Rockwell Automation. Wenn Sie eines dieser Dokumente benötigen, wenden Sie sich bitte an die Niederlassung oder den Distributor von Rockwell Automation vor Ort.

| Thema | Dokumentation | Dokumentnummer |
|---|---|---|
| Informationen zur Funktionsweise und Verwendung von Mikrosteuerungen. | MicroMentor | 1761-MMB |
| Informationen zur Montage und Verdrahtung der speicherprogrammierbaren Steuerungen MicroLogix 1200, einschließlich einer Montageschablone und Aufkleber. | Installationsanleitung für die speicherprogrammierbare Steuerung MicroLogix 1200 | 1762-IN006 |
| Ausführliche Informationen zur Planung, Montage, Verdrahtung und Fehlersuche Ihres MicroLogix 1200-Systems. | Benutzerhandbuch zur speicherprogrammierbaren Steuerung MicroLogix 1200 | 1762-UM001 |
| Informationen zur Montage und Verdrahtung der MicroLogix 1500-Basiseinheiten, einschließlich einer Montageschablone für die einfache Installation | Installationsanleitung für Basiseinheiten der speicherprogrammierbaren Steuerung MicroLogix 1500 | 1764-IN001A |
| Ausführliche Informationen zur Planung, Montage, Verdrahtung und Fehlersuche Ihres MicroLogix 1500-Systems. | Benutzerhandbuch zu der speicherprogrammierbaren Steuerung MicroLogix 1500 | 1764-UM001A |
| Beschreibung der Installation und Verkabelung eines erweiterten Schnittstellenwandlers AIC+. Dieses Handbuch enthält außerdem Informationen zur Verdrahtung von Netzwerken. | Advanced Interface Converter (AIC+) User Manual | 1761-6.4 |
| Informationen zur Installation, Konfiguration und Inbetriebnahme einer DNI | DeviceNet™ Interface User Manual | 1761-6.5 |
| Informationen zum offenen DF1-Protokoll | DF1 Protocol and Command Set Reference Manual | 1770-6.5.16 |
| Detaillierte Informationen zur Erdung und Verdrahtung speicherprogrammierbarer Steuerungen von Allen-Bradley | Richtlinien zur störungsfreien Verdrahtung und Erdung von industriellen Automatisierungsprogrammen | 1770-4.1 |
| Beschreibung der wichtigsten Unterschiede zwischen elektronischen speicherprogrammierbaren Steuerungen und festverdrahteten elektromechanischen Geräten | Application Considerations for Solid-State Controls | SGI-1.1 |
| Artikel über die Drahtstärken und -typen für die Erdung elektrischer Geräte | National Electrical Code – Veröffentlicht von der National Fire Protection Association, Boston, MA. | |
| Komplette Aufstellung der aktuellen Dokumentation, einschließlich Bestellinformationen, mit Angaben zur Verfügbarkeit auf CD-ROM bzw. in anderen Sprachen | Allen-Bradley Publication Index | SD499 |
| Glossar mit Begriffen und Abkürzungen der industriellen Automatisierungstechnik | Glossar der industriellen Automatisierung von Rockwell Automation | AG-7.1 |

## Unterstützung durch Rockwell Automation

Wir bitten Sie, die in dieser Publikation enthaltenen Informationen zur Fehlersuche zu Rate zu ziehen, bevor Sie sich an Rockwell Automation wenden, um technische Unterstützung anzufordern.

Falls das Problem weiterhin bestehen sollte, wenden Sie sich an den Distributor vor Ort oder setzen Sie sich mit Rockwell Automation auf einem der folgenden Wege in Verbindung:

| | | |
|---|---|---|
| **Telefon** | USA/Kanada | 1.440.646.5800 |
| | Außerhalb der USA/Kanadas | Die Telefonnummer für Ihr Land finden Sie im Internet:<br>1. Gehen Sie zu http://www.ab.com.<br>2. Klicken Sie auf *Product Support* (http://support.automation.rockwell.com).<br>3. Klicken Sie unter *Support Centers* auf *Contact Information*. |
| **Internet** | ⇒ | 1. Gehen Sie zu http://www.ab.com.<br>2. Klicken Sie auf *Product Support* (http://support.automation.rockwell.com). |

**Kapitel 1**

**E/A-Konfiguration**

Integrierte E/A. . . . . 1-1
MicroLogix 1200-Erweiterungs-E/A . . . . . 1-3
Speicherbelegung für MicroLogix 1200-Erweiterungs-E/A . . . . . 1-4
MicroLogix 1500 Compact™-Erweiterungs-E/A . . . . . 1-9
Speicherbelegung für MicroLogix 1500 Compact™-Erweiterungs-E/A . . . . . 1-11
E/A-Adressierung . . . . . 1-21
E/A-Forcen . . . . . 1-22
Eingangsfilter . . . . . 1-22
Impulsspeicher-Eingänge. . . . . 1-23
Konfiguration der Erweiterungs-E/A mit RSLogix 500 . . . . . 1-26

**Kapitel 2**

**Speicher der Steuerung und Filetypen**

Speicher der Steuerung. . . . . 2-2
Datenfiles . . . . . 2-7
Datenfiles beim Herunterladen schützen . . . . . 2-8
Statischer Fileschutz. . . . . 2-10
Kennwortschutz. . . . . 2-11
Speicher der Steuerung löschen. . . . . 2-12
Zukünftigen Zugriff zulassen (OEM-Sperre) . . . . . 2-13

**Kapitel 3**

**Funktionsfiles**

Überblick . . . . . 3-2
Echtzeituhr-Funktionsfile . . . . . 3-3
RTA-Befehl (Echtzeituhr anpassen) . . . . . 3-5
Funktionsfile mit Einstellpotentiometerdaten . . . . . 3-6
Funktionsfile mit Speichermoduldaten . . . . . 3-7
DAT-Funktionsfile (nur MicroLogix 1500) . . . . . 3-10
Basis-Hardware-Information-Funktionsfile . . . . . 3-13
Kommunikations-Status-File. . . . . 3-14
Ein-/Ausgangsstatusfile. . . . . 3-19

**Kapitel 4**

**Programmierbefehle – Übersicht**

Befehlssatz . . . . . 4-1
Befehlsbeschreibungen verwenden . . . . . 4-2

**Kapitel 5**

**Verwenden des Hochgeschwindigkeitszählers und des programmierbaren Endschalters**

Hochgeschwindigkeitszähler – Übersicht . . . . . 5-1
Programmierbarer Endschalter – Übersicht . . . . . 5-1
HSC-Funktionsfile (Hochgeschwindigkeitszähler). . . . . 5-2
Zusammenfassung der Unterelemente des HSC-Files . . . . . 5-4
Unterelemente des HSC-Funktionsfiles . . . . . 5-5
HSL – Hochgeschwindigkeitszähler laden . . . . . 5-26
RAC – Istwert zurücksetzen . . . . . 5-27
Programmierbarer Endschalter-File (PLS) . . . . . 5-28

**Kapitel 6**

**Verwenden von Hochgeschwindigkeitsausgängen**

PTO – Pulse Train Output (Impulsausgang) . . . . . 6-1
PTO-Funktion (Impulsausgang) . . . . . 6-2
PTO-Funktionsfile (Impulsausgang) . . . . . 6-6
Zusammenfassung der Unterelemente des PTO-Files . . . . . 6-7
PWM – Pulsweitenmodulation . . . . . 6-19
PWM-Funktion . . . . . 6-19
Funktionsfile für Pulsweitenmodulation (PWM) . . . . . 6-20
Zusammenfassung der Elemente des PWM-Files . . . . . 6-21

**Kapitel 7**

**Relaisbefehle (Bitbefehle)**

XIC – Auf geschlossen prüfen; XIO – Auf offen prüfen . . . . . 7-1
OTE – Ausgang einschalten . . . . . 7-3
OTL – Ausgang verriegeln; OTU – Ausgang entriegeln . . . . . 7-4
ONS – Einzelimpuls . . . . . 7-5
OSR – Steigender Einzelimpuls; OSF – Abfallender Einzelimpuls . . . . . 7-6

**Kapitel 8**

**Zeitwerk- und Zählerbefehle**

Zeitwerkbefehle – Übersicht . . . . . 8-1
TON – Timer-Einschaltverzögerung . . . . . 8-4
TOF – Timer-Ausschaltverzögerung . . . . . 8-5
RTO – Speichernder Timer Ein . . . . . 8-6
Funktionsweise von Zählern . . . . . 8-7
CTU – Aufwärtszählung; CTD – Abwärtszählung . . . . . 8-9
RES – Zurücksetzen . . . . . 8-10

**Kapitel 9**

**Vergleichsbefehle**

Vergleichsbefehle verwenden . . . . . 9-2
EQU – Gleich; NEQ – Ungleich . . . . . 9-3
GRT – Größer als; LES – Kleiner als . . . . . 9-4
GEQ – Größer als oder gleich; LEQ – Kleiner als oder gleich . . . . . 9-5
MEQ – Maskierter Vergleich auf gleich . . . . . 9-6
LIM – Grenzwerttest . . . . . 9-7

**Kapitel 10**

**Mathematische Befehle**

Mathematische Befehle verwenden . . . . . 10-2
Aktualisierung der mathematischen Status-Bits . . . . . 10-3
Fließkomma-Datenfile (F) verwenden . . . . . 10-4
ADD – Addition; SUB – Subtraktion . . . . . 10-7
MUL – Multiplikation; DIV – Division . . . . . 10-8
NEG – Negation . . . . . 10-9
CLR – Löschen . . . . . 10-9
ABS – Absolutwert . . . . . 10-10
SCL – Skalierung . . . . . 10-12
SCP – Skalierung mit Parametern . . . . . 10-13
SQR – Quadratwurzel . . . . . 10-15

**Konvertierungsbefehle**

## Kapitel 11

Kodier- und Dekodierbefehle verwenden . . . . . 11-1
DCD – 4 in 1 auf 16 dekodieren . . . . . 11-2
ENC – Kodierung 1 auf 16 in 4 . . . . . 11-3
FRD – BCD in Ganzzahl . . . . . 11-4
TOD – In BCD . . . . . 11-8
GCD – Gray-Code. . . . . 11-10

**Logikbefehle**

## Kapitel 12

Logikbefehle verwenden . . . . . 12-1
Aktualisierung der mathematischen Status-Bits . . . . . 12-2
AND – Logisches UND . . . . . 12-3
OR – Logisches ODER. . . . . 12-4
XOR – Exklusives ODER. . . . . 12-5
NOT – Logisches NICHT . . . . . 12-6

**Verschiebebefehle**

## Kapitel 13

MOV – Verschieben . . . . . 13-1
MVM – Maskierte Verschiebung. . . . . 13-3

**Filebefehle**

## Kapitel 14

CPW – Wort kopieren . . . . . 14-2
COP – File kopieren . . . . . 14-4
FLL – Filefüllung . . . . . 14-5
BSL – Bit nach links verschieben . . . . . 14-6
BSR – Bit nach rechts verschieben . . . . . 14-8
FFL – FIFO laden . . . . . 14-10
FFU – FIFO entladen . . . . . 14-13
LFL – LIFO laden . . . . . 14-15
LFU – LIFO entladen . . . . . 14-17
SWP – Byte-Tausch . . . . . 14-19

**Schrittschaltwerksbefehle**

## Kapitel 15

SQC – Schrittschaltwerksvergleich . . . . . 15-2
SQO – Schrittschaltwerksausgang. . . . . 15-6
SQL – Schrittschaltwerksladung . . . . . 15-9

**Programmsteuerungsbefehle**

## Kapitel 16

JMP – Sprung zu Marke. . . . . 16-1
LBL – Marke . . . . . 16-2
JSR – Sprung ins Unterprogramm. . . . . 16-2
SBR – Unterprogramm . . . . . 16-3
RET – Rückkehr vom Unterprogramm . . . . . 16-3
SUS – Suspend . . . . . 16-4
TND – Temporäres Ende . . . . . 16-4
END – Programmende . . . . . 16-5
MCR – Hauptsteuerbefehl. . . . . 16-5

## Kapitel 17

**Ein- und Ausgangsbefehle**

IIM – Sofortiger Eingang mit Maske . . . . . 17-1
IOM – Sofortiger Ausgang mit Maske . . . . . 17-4
REF – E/A auffrischen . . . . . 17-5

## Kapitel 18

**Interrupts verwenden**

Informationen zur Verwendung von Interrupts . . . . . 18-2
Anwender-Interrupt-Befehle . . . . . 18-7
INT – Interrupt-Unterprogramm . . . . . 18-7
STS – STI starten . . . . . 18-8
UID – Anwender-Interrupt deaktivieren . . . . . 18-9
UIE – Anwender-Interrupt aktivieren . . . . . 18-10
UIF – Anwender-Interrupt entfernen . . . . . 18-11
STI-Funktionsfile verwenden . . . . . 18-12
EII-Funktionsfile verwenden . . . . . 18-17

## Kapitel 19

**Prozesssteuerungsbefehl**

Konzept des PID-Befehls . . . . . 19-1
PID-Gleichung . . . . . 19-2
PD-Datenfile . . . . . 19-2
PID – Proportional-/Integral-/Differenzialverhalten . . . . . 19-3
Eingangsparameter . . . . . 19-4
Ausgangsparameter . . . . . 19-7
Abstimmparameter . . . . . 19-8
Laufzeitfehler . . . . . 19-17
Analog-E/A-Skalierung . . . . . 19-18
Anwendungshinweise . . . . . 19-19
Anwendungsbeispiele . . . . . 19-23

## Kapitel 20

**ASCII-Befehle**

Allgemeine Informationen . . . . . 20-1
ASCII-Befehle . . . . . 20-1
Befehlstypen und -funktionsweise . . . . . 20-2
Übersicht zu den Protokollen . . . . . 20-4
ST-Datenfile (String; Zeichenkette) . . . . . 20-5
Steuerdatenfile . . . . . 20-6
ACL – ASCII-Puffer löschen . . . . . 20-7
AIC – ASCII-Ganzzahl in Zeichenkette . . . . . 20-8
AWA – ASCII schreiben und anhängen . . . . . 20-9
AWT – ASCII schreiben . . . . . 20-12
ABL – Puffer auf Zeile überprüfen . . . . . 20-14
ACB – Anzahl der ASCII-Zeichen im Puffer . . . . . 20-16
ACI – Zeichenkette in Ganzzahl . . . . . 20-17
ACN – Zeichenkette verketten . . . . . 20-19
AEX – Zeichenkette extrahieren . . . . . 20-20
AHL – ASCII-Handshake-Leitungen . . . . . 20-21
ARD – ASCII-Lesen . . . . . 20-23
ARL – ASCII-Zeile lesen . . . . . 20-24
ASC – Zeichenkette suchen . . . . . 20-26

ASR – ASCII-Zeichenkette vergleichen . . . . . 20-28
Zeitdiagramm für die ARD-, ARL-, AWA- und AWT-Befehle . . . . . 20-29
Verwenden der eingeschleiften indirekten Adressierung . . . . . 20-30
Fehlercodes zu ASCII-Befehlen . . . . . 20-31
ASCII-Zeichensatz . . . . . 20-32

## Kapitel 21

**Kommunikationsbefehle**

Nachrichtenfunktion – Übersicht . . . . . 21-1
SVC – Kommunikationsbearbeitung . . . . . 21-3
MSG – Nachricht . . . . . 21-5
Nachrichten-Element . . . . . 21-6
Zeitdiagramm für MSG-Befehl . . . . . 21-12
Kontaktplanlogik für MSG-Befehl . . . . . 21-15
Zentrale Nachrichten . . . . . 21-16
Zentrale Nachrichtenübertragung konfigurieren . . . . . 21-18
Beispiele für zentrale Nachrichten . . . . . 21-25
Dezentrale Nachrichten . . . . . 21-37
Dezentrale Nachrichtenübertragung konfigurieren . . . . . 21-39
Fehlercodes zu MSG-Befehlen . . . . . 21-43

## Kapitel 22

**Rezept (nur MicroLogix 1500) und Datenprotokollierung (nur MicroLogix 1500 1764-LRP-Prozessoren)**

RCP – Rezept (nur MicroLogix 1500) . . . . . 22-1
Datenprotokollierung . . . . . 22-7
Warteschlangen und Datensätze . . . . . 22-7
Konfigurieren von Datenprotokoll-Warteschlangen . . . . . 22-11
DLG – Datenprotokollbefehl . . . . . 22-13
Datenprotokoll-Statusfile . . . . . 22-14
Laden (Lesen) von Datensätzen . . . . . 22-16
Zugreifen auf den Ladefile . . . . . 22-16
Bedingungen, die mit dem Datenladefile auftreten . . . . . 22-18

## Anhang A

**MicroLogix 1200 – Speicherbelegung und Befehlsausführungszeit**

Speicherbelegung und Ausführungszeit von Programmierbefehlen A-1
MicroLogix 1200 – Arbeitsblatt zur Abfragezeit . . . . . A-7

## Anhang B

**MicroLogix 1500 – Speicherbelegung und Befehlsausführungszeit**

Speicherbelegung und Ausführungszeit von Programmierbefehlen B-1
MicroLogix 1500 – Arbeitsblatt zur Abfragezeit . . . . . B-7

**Anhang C**

**Systemstatusfile**

Übersicht Statusfile . . . . . C-2
Details des Statusfiles. . . . . C-3

**Anhang D**

**Fehlermeldungen und Fehlercodes**

Erkennen von Steuerungsfehlern . . . . . D-1
Unterstützung durch Rockwell Automation. . . . . D-11

**Anhang E**

**Protokollkonfiguration**

DH-485-Kommunikationsprotokoll. . . . . E-2
DF1-Vollduplex-Protokoll. . . . . E-5
DF1-Halbduplex-Protokoll . . . . . E-6
Modbus™-RTU-Slave-Protokoll (nur MicroLogix 1200-Steuerungen und MicroLogix 1500-Prozessoren der Serie B und höher) . . . . . E-9
ASCII-Treiber (nur MicroLogix 1200- und 1500-Steuerungen der Serie B und höher) . . . . . E-15

**Glossar**

**Index**

**MicroLogix 1200 und 1500 Alphabetische Liste der Befehle**

# E/A-Konfiguration

In diesem Kapitel werden die Merkmale der Ein- und Ausgänge der MicroLogix 1200- und MicroLogix 1500-Steuerung beschrieben. Jede Steuerung wird mit einer bestimmten Menge *eingebetteter E/A* geliefert, die sich physisch an der Steuerung befinden. Außerdem können der Steuerung *Erweiterungs-E/A* hinzugefügt werden.

In diesem Kapitel werden folgende E/A-Funktionen erläutert:


- „Integrierte E/A“ auf Seite 1-1
- „MicroLogix 1200-Erweiterungs-E/A“ auf Seite 1-3
- „Speicherbelegung für MicroLogix 1200- Erweiterungs-E/A“ auf Seite 1-4
- „MicroLogix 1500 Compact™- Erweiterungs-E/A“ auf Seite 1-9
- „Speicherbelegung für MicroLogix 1500 Compact™- Erweiterungs-E/A“ auf Seite 1-11
- „E/A-Adressierung“ auf Seite 1-21
- „E/A-Forcen“ auf Seite 1-22
- „Eingangsfilter“ auf Seite 1-22
- „Impulsspeicher-Eingänge“ auf Seite 1-23


## Integrierte E/A

In der folgenden Tabelle sind die diskreten E/A aufgeführt, die in die Steuerungen MicroLogix 1200 und 1500 eingebaut sind. Diese E/A-Punkte werden als integrierte E/A bezeichnet.

| Steuerungsfamilie | | Eingänge | | Ausgänge | |
|---|---|---|---|---|---|
| | | Anzahl | Typ | Anzahl | Typ |
| MicroLogix 1200-Steuerungen | 1762-L24BWA | 14 | 24 V DC | 10 | Relais |
| | 1762-L24AWA | 14 | 120 V AC | 10 | Relais |
| | 1762-L24BXB | 14 | 24 V DC | 10 | 5 Relais<br>5 FET |
| | 1762-L40BWA | 24 | 24 V DC | 16 | Relais |
| | 1762-L40AWA | 24 | 120 V AC | 16 | Relais |
| | 1762-L40BXB | 24 | 24 V DC | 16 | 8 Relais<br>8 FET |
| MicroLogix 1500-Basisgeräte | 1764-24BWA | 12 | 24 V DC | 12 | Relais |
| | 1764-24AWA | 12 | 120 V AC | 12 | Relais |
| | 1764-28BXB | 16 | 24 V DC | 12 | 6 Relais<br>6 FET |

Integrierte AC-Eingänge verfügen über feste Eingangsfilter. Integrierte DC-Eingänge verfügen über konfigurierbare Eingangsfilter, die ent- sprechend der konkreten Anwendung für eine Reihe von Sonderfunk- tionen eingestellt werden können. Hierzu gehören: Hochgeschwindig- keitszählung, Ereignis-Interrupts und Impulsspeichereingänge. Die 1764-28BXB verfügt über zwei Hochgeschwindigkeits-Ausgänge, die als PTO-Ausgänge (Pulse Train Output) und/oder als PWM-Ausgänge (Pulse Width Modulated) eingesetzt werden können. Die 1762- L24BXB- und 1762-L40BXB-Steuerungen verfügen jeweils über einen Hochgeschwindigkeits-Ausgang.

# MicroLogix 1200-Erweiterungs-E/A

Falls die Anwendung mehr E/A erfordert, als die Steuerung zur Verfügung stellt, können E/A-Module angeschlossen werden. Diese Module werden als *Erweiterungs-E/A* bezeichnet.

## Erweiterungs-E/A-Module

Mit den MicroLogix 1200-Erweiterungs-E/A (Bulletin 1762) werden diskrete und analoge Ein- und Ausgänge und zukünftig auch Spezialmodule bereitgestellt. Bei der MicroLogix 1200-Steuerung können bis zu sechs zusätzliche E/A-Module angeschlossen werden. Die Anzahl der 1762-E/A-Module, die an die MicroLogix 1200- Steuerung angeschlossen werden können, hängt von der durch die E/A-Module aufgenommenen Leistung ab.

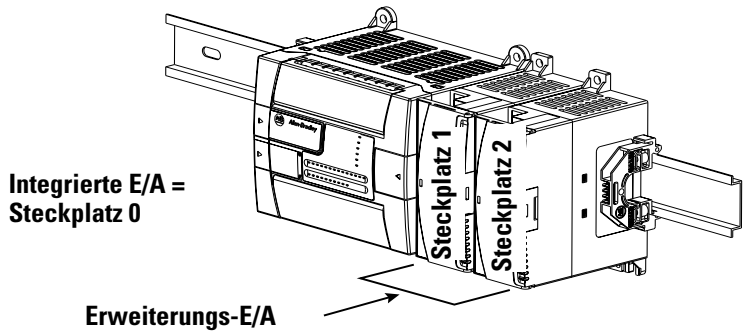
Weitere Informationen zur gültigen Konfiguration finden Sie im Benutzerhandbuch *Speicherprogrammierbare Steuerungen MicroLogix 1200*, Publikation 1762-UM001.

**HINWEIS** Eine Zuordnungstabelle für MicroLogix 1200-Erweiterungs-E/A finden Sie auf der MicroLogix- Website (http://www.ab.com/micrologix).

## Adressierung der Erweiterungs-E/A-Steckplätze

Die nachfolgende Abbildung zeigt, wie die MicroLogix 1200-Steuerung und ihre E/A adressiert werden.

Die Erweiterungs-E/A werden als Steckplätze 1 bis 6 adressiert (die integrierte E/A der Steuerung wird als Steckplatz 0 adressiert). Die Zählung der Module erfolgt von links nach rechts (siehe unten).




**HINWEIS** In den meisten Fällen können Sie das folgende Adressformat verwenden:
X:s/b (X = Buchstabe für Filetyp,
s = Steckplatznummer, b = Bitnummer)

Ausführliche Informationen zu Adressformaten finden Sie im Abschnitt „E/A-Adressierung“ auf Seite 1-21.

# Speicherbelegung für MicroLogix 1200-Erweiterungs-E/A

## Diskrete E/A-Konfiguration

### *1762-IA8- und 1762-IQ8-Eingangsdaten*

Der Eingangsdatenfile enthält zu jedem Eingangsmodul den aktuellen Status der Feldeingangspunkte. Die Bitpositionen 0 bis 7 entsprechen den Eingangsklemmen 0 bis 7.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | x | x | x | x | x | x | x | x | r | r | r | r | r | r | r | r |

r = Nur lesen, x = nicht verwendet, immer 0 oder im AUS-Zustand

### *1762-IQ16-Eingangsdaten*

Der Eingangsdatenfile enthält zu jedem Eingangsmodul den aktuellen Status der Feldeingangspunkte. Die Bitpositionen 0 bis 15 entsprechen den Eingangsklemmen 0 bis 15.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

r = Nur lesen

### *1762-OA8, 1762-OB8 und 1762-OW8-Ausgangsdaten*

Für jedes Ausgangsmodul enthält der Ausgangsdatenfile den von der Steuerung zugewiesenen Status der diskreten Ausgangspunkte. Die Bitpositionen 0 bis 7 entsprechen den Ausgangsklemmen 0 bis 7.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

r/w = lesen und schreiben, 0 = immer 0 oder im AUS-Zustand

### *1762-OB16 und 1762-OW16-Ausgangsdaten*

Für jedes Ausgangsmodul enthält der Ausgangsdatenfile den von der Steuerung zugewiesenen Status der diskreten Ausgangspunkte. Die Bitpositionen 0 bis 15 entsprechen den Ausgangsklemmen 0 bis 15.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

r/w = lesen und schreiben

## Analog-E/A-Konfiguration

In der folgenden Tabelle werden die Datenbereiche von 0 bis 10 V DC und 4 bis 20 mA angezeigt.

**Tabelle 1.1 Gültige Eingangs-/Ausgangs-Datenwortformate/-bereiche**

| Normaler Betriebsbereich | Maximalbereich | Roh-/Proportionaldaten | Skaliert für PDI |
|---|---|---|---|
| 0 bis 10 V DC | 10,5 V DC | 32760 | 16380 |
| | 0,0 V DC | 0 | 0 |
| 4 bis 20 mA | 21,0 mA | 32760 | 16380 |
| | 20,0 mA | 31200 | 15600 |
| | 4,0 mA | 6240 | 3120 |
| | 0,0 mA | 0 | 0 |

### *1762-IF2OF2-Eingangsdatenfile*

Zu jedem Eingangsmodul enthalten die Wörter 0 und 1 zu Steckplatz x die Analogwerte der Eingänge. Das Modul kann für die Verwendung der Roh-/Proportionaldaten oder der für PID skalierten Daten konfiguriert werden. Die Eingangsdatenfiles für die verschiedenen Konfigurationen sind nachfolgend dargestellt.

**Tabelle 1.2 Roh-/Proportionalformat**

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | Kanal 0, Daten 0 bis 32768 | | | | | | | | | | | | 0 | 0 | 0 |
| 1 | 0 | Kanal 1, Daten 0 bis 32768 | | | | | | | | | | | | 0 | 0 | 0 |
| 2 | reserviert | | | | | | | | | | | | | | | |
| 3 | reserviert | | | | | | | | | | | | | | | |
| 4 | reserviert | | | | | | | | | | | | | | S1 | S0 |
| 5 | U0 | O0 | U1 | O1 | reserviert | | | | | | | | | | | |

**Tabelle 1.3 Skaliert für PID**

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | Kanal 0, Daten 0 bis 16383 | | | | | | | | | | | | 0 | 0 |
| 1 | 0 | 0 | Kanal 1, Daten 0 bis 16383 | | | | | | | | | | | | 0 | 0 |
| 2 | reserviert | | | | | | | | | | | | | | | |
| 3 | reserviert | | | | | | | | | | | | | | | |
| 4 | reserviert | | | | | | | | | | | | | | S1 | S0 |
| 5 | U0 | O0 | U1 | O1 | reserviert | | | | | | | | | | | |

Die Bits sind wie folgt definiert:

- Sx = Allgemeine Status-Bits für die Kanäle 0 und 1. Dieses Bit wird gesetzt, wenn ein Fehler (Bereichsüber- oder -unterschreitung) für den betreffenden Kanal oder ein allgemeiner Hardwarefehler im Modul aufgetreten ist.
- Ox = Bereichsüberschreitungs-Flag-Bits für die Kanäle 0 und 1. Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden.
- Ux = Bereichsunterschreitungs-Flag-Bits für die Kanäle 0 und 1. Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden.

### *1762-IF2OF2-Ausgangsdatenfile*

Für jedes Modul enthalten die Wörter 0 und 1, Steckplatz x, die Kanalausgangsdaten.

**Tabelle 1.4 Roh-/Proportionalformat**

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | Kanal 0, Daten 0 bis 32768 | | | | | | | | | | | | 0 | 0 | 0 |
| 1 | 0 | Kanal 1, Daten 0 bis 32768 | | | | | | | | | | | | 0 | 0 | 0 |

**Tabelle 1.5 Skaliert für PID**

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | Kanal 0, Daten 0 bis 16383 | | | | | | | | | | | | 0 | 0 |
| 1 | 0 | 0 | Kanal 1, Daten 0 bis 16383 | | | | | | | | | | | | 0 | 0 |

### *1762-IF4-Eingangsdatenfile*

Zu jedem Eingangsmodul enthalten die Wörter 0 und 1 zu Steckplatz x die Analogwerte der Eingänge. Das Modul kann für die Verwendung der Roh-/Proportionaldaten oder der für PID skalierten Daten konfiguriert werden. Die Eingangsdatenfiles für die verschiedenen Konfigurationen sind nachfolgend dargestellt.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | SGN0 | Daten, Kanal 0 | | | | | | | | | | | | | | |
| 1 | SGN1 | Daten, Kanal 1 | | | | | | | | | | | | | | |
| 2 | SGN2 | Daten, Kanal 2 | | | | | | | | | | | | | | |
| 3 | SGN3 | Daten, Kanal 3 | | | | | | | | | | | | | | |
| 4 | reserviert | | | | | | | | | | | | S3 | S2 | S1 | S0 |
| 5 | U0 | O0 | U1 | O1 | U2 | O2 | U3 | O3 | reserviert | | | | | | | |
| 6 | reserviert | | | | | | | | | | | | | | | |

Die Bits sind wie folgt definiert:

- Sx = Allgemeine Status-Bits für die Kanäle 0 bis 3. Dieses Bit wird gesetzt, wenn ein Fehler (Bereichsüber- oder -unterschreitung) für den betref- fenden Kanal oder ein allgemeiner Hardwarefehler im Modul aufgetreten ist.
- Ox = Bereichsüberschreitungs-Flag-Bits für die Kanäle 0 bis 3. Die Bits werden gesetzt, wenn das Eingangssignal oberhalb des benutzer- definierten Bereichs liegt. Das Modul fährt während einer Bereichsüber- schreitung fort, Daten bis zum maximalen Bereichswert zu konvertieren. Die Bits werden zurückgesetzt, wenn die Bereichsüberschreitung nicht mehr vorliegt.
- UIx = Bereichsunterschreitungs-Flag-Bits für die Eingangskanäle 0 bis 3. Die Bits werden gesetzt, wenn das Eingangssignal unterhalb des benutzer- definierten Bereichs liegt. Das Modul fährt während einer Bereichsunter- schreitung fort, Daten bis zum maximalen Bereichswert zu konvertieren. Die Bits werden zurückgesetzt, wenn die Bereichsunterschreitung nicht mehr vorliegt.
- SGNx = Das Vorzeichen-Bit für die Kanäle 0 bis 3.

## Spezielle E/A-Konfiguration

*Eingangsdatenfile für 1762-IR4-RTD-/Widerstandsmodul*

Zu jedem Modul enthalten die Wörter 0 bis 3 zu Steckplatz x die Analogwerte der Eingänge. Wörter 4 und 5 liefern Statusfeedback für Sensor/Kanal. Die Eingangsdatenfiles für die verschiedenen Konfigurationen sind nachfolgend dargestellt.

| Wort/ Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Analogeingangsdaten, Kanal 0 | | | | | | | | | | | | | | | |
| 1 | Analogeingangsdaten, Kanal 1 | | | | | | | | | | | | | | | |
| 2 | Analogeingangsdaten, Kanal 2 | | | | | | | | | | | | | | | |
| 3 | Analogeingangsdaten, Kanal 3 | | | | | | | | | | | | | | | |
| 4 | Reserviert | | | | OC3 | OC2 | OC1 | OC0 | Reserviert | | | | S3 | S2 | S1 | S0 |
| 5 | U0 | O0 | U1 | O1 | U2 | O2 | U3 | O3 | Reserviert | | | | | | | |

Die Bits sind wie folgt definiert:

- Sx = Allgemeine Status-Bits für die Eingangskanäle 0 bis 3. Dieses Bit wird gesetzt (1), wenn ein Fehler (Bereichsüber- oder -unter- schreitung, offener Schaltkreis oder ungültige Eingangsdaten) für den betreffenden Kanal oder ein allgemeiner Hardwarefehler im Modul aufgetreten ist. Ein Zustand nicht gültiger Eingangsdaten wird vom Anwenderprogramm erkannt. Nähere Angaben finden Sie im Benutzerhandbuch zum MicroLogix™ 1200 RTD-/Wider- standseingangsmodule, Publikation 1762-UM003.
- OCx = Hinweis auf einen offenen Schaltkreis für Kanäle 0 bis 3 mithilfe der RTD- oder Widerstandseingänge-Kurzschlusserken- nung nur für RTD-Eingänge. Kurzschlusserkennung für Wider- standseingänge ist nicht angegeben, da 0 ein gültiger Wert ist.
- Ox = Bereichsüberschreitungs-Flag-Bits für Eingangskanäle 0 bis 3 mithilfe der RTD- oder Widerstandseingänge. Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden.
- Ux = Bereichsunterschreitungs-Flag-Bits für Kanäle 0 bis 3 nur mithilfe der RTD-Eingänge.
  Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden. Bereichsunterschreitungserkennung für direkte Widerstandseingänge ist nicht angegeben, da 0 ein gültiger Wert ist.

### *Eingangsdatenfile für 1762-IT4-Thermoelementmodul*

Zu jedem Modul enthalten die Wörter 0 bis 3 zu Steckplatz x die Analogwerte der Eingänge. Im Folgenden wird ein Eingangsdatenfile gezeigt.

| **Wort/ Bit** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | SGN | Analogeingangsdaten, Kanal 0 | | | | | | | | | | | | | | |
| 1 | SGN | Analogeingangsdaten, Kanal 1 | | | | | | | | | | | | | | |
| 2 | SGN | Analogeingangsdaten, Kanal 2 | | | | | | | | | | | | | | |
| 3 | SGN | Analogeingangsdaten, Kanal 3 | | | | | | | | | | | | | | |
| 4 | Reserviert | | | OC4 | OC3 | OC2 | OC1 | OC0 | Reserviert | | | S4 | S3 | S2 | S1 | S0 |
| 5 | U0 | O0 | U1 | O1 | U2 | O2 | U3 | O3 | U4 | O4 | Reserviert | | | | | |

Die Bits sind wie folgt definiert:

- Sx = Allgemeine Status-Bits für die Kanäle 0 bis 3 (S0 bis S3) und den CJC-Sensor (S4). Dieses Bit wird gesetzt (1), wenn ein Fehler (Bereichsüber- oder -unterschreitung, offener Schaltkreis oder ungültige Eingangsdaten) für den betreffenden Kanal aufgetreten ist. Ein Zustand nicht gültiger Eingangsdaten wird vom Anwender- programm erkannt. Nähere Angaben finden Sie im Benutzer- handbuch *MicroLogix™ 1200 I/O Thermocouple/mV Input Module User Manual* (Publikation 1762-UM002).
- OCx = Hinweis auf einen offenen Schaltkreis für Kanäle 0 bis 3 (OC0 bis OC3) und den CJC-Sensor (OC4).
- Ox = Bereichsüberschreitungs-Flag-Bits für Kanäle 0 bis 3 (O0 bis O3) und den CJC-Sensor (O4). Diese Bits können im Steuerungs- programm zur Fehlererkennung verwendet werden.
- Ux = Bereichsunterschreitungs-Flag-Bits für Kanäle 0 bis 3 (U0 bis U3) und den CJC-Sensor (U4). Diese Bits können im Steuerungs- programm zur Fehlererkennung verwendet werden.

# MicroLogix 1500 Compact™- Erweiterungs-E/A

Falls die Anwendung mehr E/A erfordert, als die Steuerung zur Verfügung stellt, können E/A-Module angeschlossen werden. Diese Module werden als *Erweiterungs-E/A* bezeichnet.

## Erweiterungs-E/A-Module

Mit Compact-E/A (Bulletin 1769) werden diskrete und analoge Ein- und Ausgänge und zukünftig auch Spezialmodule bereitgestellt. Bei der MicroLogix 1500-Steuerung können bis zu 16(1) zusätzliche E/A-Module angeschlossen werden. Die Anzahl der anschließbaren Module hängt von der Leistung ab, die von den E/A-Modulen aufgenommen wird.

Weitere Informationen zur gültigen Konfiguration finden Sie im Benutzerhandbuch *Speicherprogrammierbare Steuerungen MicroLogix 1500*, Publikation 1764-UM001.

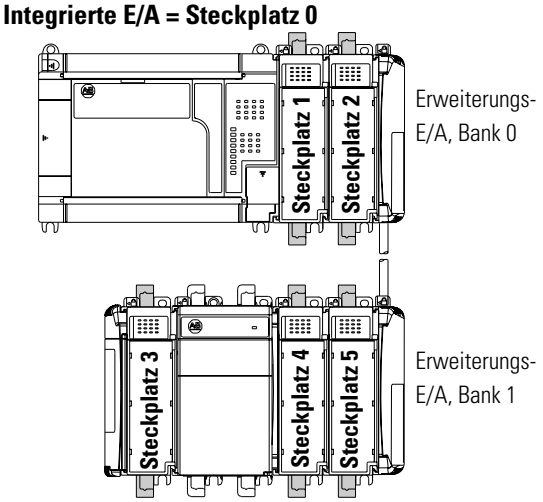
**HINWEIS** Eine Zuordnungstabelle für MicroLogix 1500-Erweiterungs-E/A finden Sie auf der MicroLogix- Website (http://www.ab.com/micrologix).

## Erweiterungs-E/A-Adressierung

Die nachfolgende Abbildung zeigt, wie die MicroLogix 1500- Steuerung und ihre E/A adressiert werden.

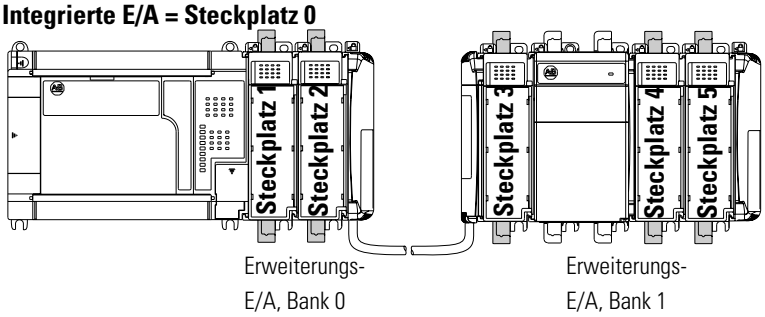
Die Erweiterungs-E/A werden als Steckplätze 1 bis 16 adressiert (die integrierte E/A der Steuerung wird als Steckplatz 0 adressiert). Netzteile und -kabel werden nicht als Steckplätze gezählt, sondern müssen dem RSLogix 500-Projekt in der E/A-Konfiguration hinzu- gefügt werden. Die Module werden auf jeder Bank von links nach rechts gezählt (siehe nachfolgende Abbildungen).

**Abbildung 1.1 Vertikale Ausrichtung**




(1) Maximal 8 Module für Basisgeräte der Serie A.

**Abbildung 1.2 Horizontale Ausrichtung**




**HINWEIS** In den meisten Fällen können Sie das folgende Adressformat verwenden:
X:s/b (X = Buchstabe für Filetyp,
s = Steckplatznummer, b = Bitnummer)

Ausführliche Informationen zu Adressformaten finden Sie im Abschnitt „E/A-Adressierung“ auf Seite 1-21.

## Erweiterungsnetzteile und -kabel

Vergewissern Sie sich vor der Verwendung eines Erweiterungs-E/A- Netzteils 1769 mit einer MicroLogix 1500-Steuerung, dass Sie über folgende Voraussetzungen verfügen:

- MicroLogix 1500-Prozessor:
  Bestell-Nr. 1764-LSP, FRN 3 und höher
  Bestell-Nr. 1764-LRP, FRN 4 und höher
- Betriebssystemversion: Sie finden die FRN in dem Wort S:59 (Betriebssystem-FRN) in dem Statusfile.

**WICHTIG** Falls Ihr Prozessor eine ältere Revisionsnummer aufweist, müssen Sie das Betriebssystem auf FRN 3 oder höher aktualisieren, um ein Erweiterungskabel und ein Netzteil zu verwenden. Die Betriebssystem- erweiterung können Sie im Internet unter http://www.ab.com/micrologix herunterladen. Wählen Sie „MicroLogix 1500 System“, und klicken Sie auf „Downloads“.

**ACHTUNG** ES IST NUR EIN ERWEITERUNGSNETZTEIL UND -KABEL ZULÄSSIG

Das Erweiterungsnetzteil kann nicht direkt an die Steuerung angeschlossen werden. Es ist über eines der Erweiterungskabel anzuschließen. In einem MicroLogix 1500-System kann nur ein Erweiterungs- netzteil verwendet werden. Wird diese Einschränkung nicht beachtet, besteht die Gefahr der Beschädigung des Netzteils und unvorhergesehener Reaktionen.

# Speicherbelegung für MicroLogix 1500 Compact™-Erweiterungs-E/A

## Diskrete E/A-Konfiguration

### *Eingangsdaten 1769-IA8I*

Der Eingangsdatenfile enthält zu jedem Eingangsmodul den aktuellen Status der Feldeingangspunkte. Die Bitpositionen 0 bis 7 entsprechen den Eingangsklemmen 0 bis 7, die Bits 8 bis 15 werden nicht verwendet.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | x | x | x | x | x | x | x | x | r | r | r | r | r | r | r | r |

r = Lesen, x = nicht verwendet, immer 0 oder im AUS-Zustand

### *Eingangsdaten 1769-IM12*

Der Eingangsdatenfile enthält zu jedem Eingangsmodul den aktuellen Status der Feldeingangspunkte. Die Bitpositionen 0 bis 11 entsprechen den Eingangsklemmen 0 bis 11, die Bits 12 bis 15 werden nicht verwendet.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | x | x | x | x | r | r | r | r | r | r | r | r | r | r | r | r |

r = Lesen, x = nicht verwendet, immer 0 oder im AUS-Zustand

### *Eingangsdaten 1769-IA16 und 1769-IQ16*

Der Eingangsdatenfile enthält zu jedem Eingangsmodul den aktuellen Status der Feldeingangspunkte. Die Bitpositionen 0 bis 15 entsprechen den Eingangsklemmen 0 bis 15.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

r = Lesen

### *1769-IQ6XOW4-Eingangsdaten*

Der Eingangsdatenfile enthält zu jedem Modul den aktuellen Status der Feldeingangspunkte. Die Bitpositionen 0 bis 5 entsprechen den Eingangsklemmen 0 bis 5, die Bits 6 bis 15 werden nicht verwendet.

| Wort | Eingangs-Bit-Position | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | x | x | x | x | x | x | x | x | x | x | r | r | r | r | r | r |

r = Lesen, x = nicht verwendet, immer 0 oder im AUS-Zustand

### *1769-IQ6XOW4-Ausgangsdaten*

Der Ausgangsdatenfile enthält zu jedem Modul den aktuellen, durch das Steuerungsprogramm festgelegten Status der diskreten Ausgangs- punkte. Die Bitpositionen 0 bis 3 entsprechen den Ausgangsklemmen 0 bis 3, die Bits 4 bis 15 werden nicht verwendet.

| Wort | Ausgangs-Bit-Position | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | x | x | x | x | x | x | x | x | x | x | x | x | r/w | r/w | r/w | r/w |

r/w = lesen und schreiben, x = nicht verwendet, immer 0 oder im AUS-Zustand

### *Ausgangsdaten 1769-OA8, 1769-OW8 und 1769-OW8I*

Der Ausgangsdatenfile enthält zu jedem Modul den aktuellen, durch das Steuerungsprogramm festgelegten Status der diskreten Ausgangs- punkte. Die Bitpositionen 0 bis 7 entsprechen den Ausgangsklemmen 0 bis 7, die Bits 8 bis 15 werden nicht verwendet.

| Wort | Ausgangs-Bit-Position | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | x | x | x | x | x | x | x | x | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

r/w = lesen und schreiben, x = nicht verwendet, immer 0 oder im AUS-Zustand

### *Ausgangsdaten 1769-OA16, 1769-OB16, 1769-OB16P, 1769-OV16 und 1769-OW16*

Der Ausgangsdatenfile enthält zu jedem Modul den aktuellen, durch das Steuerungsprogramm festgelegten Status der diskreten Ausgangs- punkte. Die Bitpositionen 0 bis 15 entsprechen den Ausgangsklemmen 0 bis 15.

| Wort | Ausgangs-Bit-Position | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

r/w = lesen und schreiben

## Analog-E/A-Konfiguration

### *1769-IF4-Eingangsdatenfile*

Zu jedem Eingangsmodul enthalten die Wörter 0 bis 3 die Analog- werte der Eingänge.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | SGN | Analogeingangsdaten, Kanal 0 | | | | | | | | | | | | | | |
| 1 | SGN | Analogeingangsdaten, Kanal 1 | | | | | | | | | | | | | | |
| 2 | SGN | Analogeingangsdaten, Kanal 2 | | | | | | | | | | | | | | |
| 3 | SGN | Analogeingangsdaten, Kanal 3 | | | | | | | | | | | | | | |
| 4 | nicht belegt | | | | | | | | | | | | S3 | S2 | S1 | S0 |
| 5 | U0 | O0 | U1 | O1 | U2 | O2 | U3 | O3 | Auf 0 gesetzt | | | | | | | |

Die Bits sind wie folgt definiert:

- SGN = Vorzeichen-Bit im Zweier-Komplement-Format.
- Sx = Allgemeine Status-Bits für Kanäle 0 bis 3. Dieses Bit wird gesetzt (1), wenn ein Fehler (Bereichsüber- oder -unterschreitung) für diesen Kanal auftritt.
- Ux = Bereichsunterschreitungs-Flag-Bits für Kanäle 0 bis 3. Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden.
- Ox = Bereichsüberschreitungs-Flag-Bits für Kanäle 0 bis 3. Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden.

### *1769-OF2-Ausgangsdatenfile*

Für jedes Modul enthalten die Wörter 0 und 1 in dem Ausgangs- datenfile die Ausgangsdaten der Kanäle 0 und 1.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | SGN | Kanal 0, Daten 0 bis 32768 | | | | | | | | | | | | | | |
| 1 | SGN | Kanal 1, Daten 0 bis 32768 | | | | | | | | | | | | | | |

SGN = Vorzeichen-Bit im Zweier-Komplement-Format.

### *1769-IF4XOF2-Eingangsdatenfile*

Der Eingangsdatenfile ermöglicht Zugriff auf die Eingangsdaten für den Gebrauch im Steuerungsprogramm, Bereichsüberschreitungser- kennung für die Eingangs- und Ausgangskanäle sowie Ausgangs- daten-Feedback, was im Nachfolgenden näher beschrieben wird.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | SGN | Analogeingangsdaten, Kanal 0 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | SGN | Analogeingangsdaten, Kanal 1 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | SGN | Analogeingangsdaten, Kanal 2 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | SGN | Analogeingangsdaten, Kanal 3 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | nicht verwendet(1) | | | | | | | | | | | | I3 | I2 | I1 | I0 |
| 5 | nicht verwen-det | H0 | nicht verwen-det | H1 | nicht verwendet(1) | | | | | | | | E1 | E0 | O1 | O0 |
| 6 | SGN | Ausgangsdatenecho/Loopback (Prüfschleife) für Ausgangskanal 0 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | SGN | Ausgangsdatenecho/Loopback (Prüfschleife) für Ausgangskanal 1 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

(1) Alle nicht verwendeten Bits werden von dem Modul auf den Wert 0 gesetzt.

**WICHTIG** Die Eingangswörter 6 und 7 enthalten die Ausgangs- datenecho-/Loopback-Informationen für die Aus- gangskanäle 0 und 1. Die Bits 0 bis 6 und Bit 15 der Wörter 6 und 7 sollten im Steuerungsprogramm immer auf den Wert 0 gesetzt werden. Wenn sie nicht auf 0 gesetzt werden, wird vom Modul das ungültige Daten-Flag (Ex) für diesen Kanal gesetzt. Der Kanal arbeitet jedoch mit dem zuvor konvertierten Wert weiter.

Die Bits sind wie folgt definiert:

- **SGN** = Vorzeichen-Bit im Zweier-Komplement-Format. Für das Modul 1769-IF4XOF2 immer positiv (gleich Null).
- **Ix** = Bereichsüberschreitungs-Flag-Bits für die Eingangskanäle 0 bis 3. Diese Bits können im Steuerungsprogramm zur Fehlerer- kennung verwendet werden. Wenn sie auf 1 gesetzt werden, signalisieren die Bits, dass das Eingangssignal außerhalb des normalen Betriebsbereichs liegt. Das Modul konvertiert jedoch weiterhin Analogdaten in den Maximalwert. Wenn die Bedingung der Bereichsüberschreitung nicht mehr erfüllt ist, werden die Bits automatisch zurückgesetzt (0).
- **Ox** = Wort 5, Bits 0 und 1 liefern Bereichsüberschreitungserken- nung für die Ausgangskanäle 0 und 1. Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden. Wenn sie auf 1 gesetzt werden, signalisieren die Bits, dass das Ausgangssignel außerhalb des normalen Betriebsbereichs liegt. Das Modul konvertiert jedoch weiterhin Analogdaten in den Maxi- malwert. Wenn die Bedingung der Bereichsüberschreitung nicht mehr erfüllt ist, werden die Bits automatisch zurückgesetzt (0).

**HINWEIS** Eine Bereichsunterschreitung wird nicht angezeigt, da Null ein gültiger Wert ist.

- **Ex** = Wenn dieses Bit gesetzt wird (1), zeigt es an, dass ungültige Daten (beispielsweise liegt der Wert, der von der Steuerung übermittelt wurde, außerhalb des standardmäßigen Ausgangs- bereichs oder der Inkremente, z. B. 128, 256 usw.) in den Ausgangsdaten-Bits 0 bis 6 oder dem Signalbit (15) gesetzt wurden.
- **Hx** = Bits, die den letzten Zustand halten. Wenn diese Bits gesetzt werden (1), zeigen sie an, dass für den Kanal der Zustand „Letzten Status halten" auftritt.
- **Wörter 6 und 7** = Diese Wörter geben das Analogausgangs- datenecho des Analogwerts wieder, der vom Digital-/Analog- Wandler konvertiert wurde, und nicht unbedingt den elektrischen Zustand der Ausgangsklemmen. Sie geben keine gebrückten oder offenen Ausgänge an.

**WICHTIG** Nur wenn die Steuerung den Programm-Modus oder Fehlermodus unterstützt und so konfiguriert wurde, dass diese beiden Modi auch verwendet werden, muss die Prüfschleife für die Eingangs- wörter 6 und 7 verwendet werden.

*1769-IF4XOF2-Ausgangsdatenfile*

Der Ausgangsdatenfile gilt nur für Ausgangsdaten aus dem Modul, wie in der nachfolgenden Tabelle dargestellt wird.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | SGN | Analog-Ausgangsdatenkanal 0 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | SGN | Analog-Ausgangsdatenkanal 1 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**WICHTIG** Die Bits 0 bis 6 und Bit 15 der Ausgangsdaten- wörter 0 und 1 sollten im Steuerungsprogramm immer auf Null gesetzt werden. Wenn sie nicht auf
0 gesetzt werden, wird vom Modul das ungültige Daten-Flag (Ex) für diesen Kanal gesetzt. Der Kanal arbeitet jedoch mit dem zuvor konvertierten Wert weiter. Wenn ein maskierter Verschiebe-Befehl (MVM, Move with Mask) mit einer Maske 7F80 (hexa- dezimaler Wert) dazu verwendet wird, Daten zu
den Ausgangswörtern zu verschieben, kann somit verhindert werden, dass Informationen zu den Bits 0 bis 6 und dem Bit 15 geschrieben werden.

## Spezielle E/A-Konfiguration

*Eingangsdatenfile für 1769-IR6-RTD-/Widerstandsmodul*

Die ersten sechs Wörter (0 bis 5) des Eingangsdatenfiles enthalten die Analog-RTD- oder -Widerstandswerte der Eingänge. Wörter 6 und 7 liefern, wie unten dargestellt wird, das Sensor-/Kanalstatus-Feedback für das Steuerungsprogramm.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | RTD-/Widerstands-Eingangsdatenkanal 0 | | | | | | | | | | | | | | | |
| 1 | RTD-/Widerstands-Eingangsdatenkanal 1 | | | | | | | | | | | | | | | |
| 2 | RTD-/Widerstands-Eingangsdatenkanal 2 | | | | | | | | | | | | | | | |
| 3 | RTD-/Widerstands-Eingangsdatenkanal 3 | | | | | | | | | | | | | | | |
| 4 | RTD-/Widerstands-Eingangsdatenkanal 4 | | | | | | | | | | | | | | | |
| 5 | RTD-/Widerstands-Eingangsdatenkanal 5 | | | | | | | | | | | | | | | |
| 6 | nicht verwendet | | OC5 | OC4 | OC3 | OC2 | OC1 | OC0 | nicht verwendet | | S5 | S4 | S3 | S2 | S1 | S0 |
| 7 | U0 | O0 | U1 | O1 | U2 | O2 | U3 | O3 | U4 | O4 | U5 | O5 | nicht verwendet | | | |

Die Statusbits für Wörter 6 und 7 werden wie folgt beschrieben:

- **Sx** = Allgemeine Status-Bits für die Kanäle 0 bis 5. Dieses Bit wird gesetzt (1), wenn ein Fehler (Bereichsüber- oder -unterschreitung, offener Schaltkreis oder ungültige Eingangsdaten) für den betreffenden Kanal aufgetreten ist. Ein Zustand nicht gültiger Eingangsdaten wird vom Anwenderprogramm erkannt. Dieser Zustand tritt auf, wenn die erste Analog-zu-Digital-Konvertierung bei Inbetriebnahme noch immer in Gang ist oder nachdem eine neue Konfiguration an das Modul gesendet wurde. Nähere Angaben finden Sie im Benutzerhandbuch Compact™ I/O RTD-/Widerstands-Eingangsmodul, Publikation 1769-UM005.
- **OCx** = Erkennungsbit für einen offenen Schaltkreis für Kanäle 0 bis 5. Diese Bits werden gesetzt (1), wenn ein offener oder über- brückter Eingang für RTD-Eingänge oder ein offener Eingang für Widerstandseingänge erkannt wurde.

  **HINWEIS** Kurzschlusserkennung für Widerstandseingänge ist nicht angegeben, da 0 ein gültiger Wert ist.

- **Ux** = Bereichsunterschreitungs-Flag-Bits für Kanäle 0 bis 5 nur mithilfe der RTD-Eingänge. Diese Bits können im Steuerungs- programm zur Fehlererkennung verwendet werden. Es gibt keinen Bereichsunterschreitungsfehler für einen direkten Widerstandseingang, da 0 ein gültiger Wert ist.
- **Ox** = Bereichsüberschreitungs-Flag-Bits für Eingangskanäle 0 bis 5 mithilfe der RTD- oder Widerstandseingänge. Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden.

### *Eingangsdatenfile für 1769-IT6-Thermoelementmodul*

Der Eingangsdatenfile enthält die analogen Werte der Eingänge.

| Wort | Bitposition | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0 | Analogeingangsdaten, Kanal 0 | | | | | | | | | | | | | | | |
| 1 | Analogeingangsdaten, Kanal 1 | | | | | | | | | | | | | | | |
| 2 | Analogeingangsdaten, Kanal 2 | | | | | | | | | | | | | | | |
| 3 | Analogeingangsdaten, Kanal 3 | | | | | | | | | | | | | | | |
| 4 | Analogeingangsdaten, Kanal 4 | | | | | | | | | | | | | | | |
| 5 | Analogeingangsdaten, Kanal 5 | | | | | | | | | | | | | | | |
| 6 | OC7 | OC6 | OC5 | OC4 | OC3 | OC2 | OC1 | OC0 | S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 |
| 7 | U0 | O0 | U1 | O1 | U2 | O2 | U3 | O3 | U4 | O4 | U5 | O5 | U6 | O6 | U7 | O7 |

Die Bits sind wie folgt definiert:

- Sx = Allgemeines Status-Bit für die Kanäle 0 bis 5 und CJC- Sensoren (S6 und S7). Dieses Bit wird gesetzt (1), wenn ein Fehler (Bereichsüber- oder -unterschreitung, offener Schaltkreis oder ungültige Eingangsdaten) für den betreffenden Kanal aufgetreten ist. Ein Zustand nicht gültiger Eingangsdaten wird vom Anwender- programm erkannt. Dieser Zustand tritt auf, wenn die erste Analog-zu-Digital-Konvertierung noch immer in Gang ist, nachdem eine neue Konfiguration an das Modul gesendet wurde.
- OCx = Die Erkennungs-Bits für offene Schaltkreise weisen auf eine offene Eingangsschaltung der Kanäle 0 bis 5 (OC0 bis OC5) und der CJC-Sensoren CJC0 (OC6) und CJC1 (OC7) hin. Das Bit wird gesetzt (1), wenn ein offener Schaltkreis vorliegt.
- Ux = Bereichsunterschreitungs-Flag-Bits für die Kanäle 0 bis 5 und die CJC-Sensoren (U6 und U7). Bei Thermoelementeingängen wird das Bereichsunterschreitungs-Bit gesetzt, wenn bei einer Temperaturmessung ein Wert unter dem normalen Betriebsbereich des jeweiligen Thermoelementtyps gemessen wird. Bei Millivolt- eingängen weist das Bereichsunterschreitungs-Bit auf eine Span- nung hin, die unter dem normalen Betriebsbereich liegt. Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden.
- Ox = Bereichsüberschreitungs-Flag-Bits für die Kanäle 0 bis 5 und die CRC-Sensoren (O6 und O7). Bei Thermoelementeingängen wird das Bereichsüberschreitungs-Bit gesetzt, wenn bei einer Temperaturmessung ein Wert über dem normalen Betriebsbereich des jeweiligen Thermoelementtyps gemessen wird. Bei Millivolt- eingängen weist das Bereichsüberschreitungs-Bit auf eine Span- nung hin, die über dem normalen Betriebsbereich liegt. Diese Bits können im Steuerungsprogramm zur Fehlererkennung verwendet werden.

### *Ausgangsanordnung für 1769-HSC-Hochgeschwindigkeitszähler-Modul*

Die Angaben in der nachfolgenden Tabelle stellen eine kurze Übersicht der Anordnung dar. Nähere Angaben finden Sie im Benutzerhandbuch *Compact™ High Speed Counter Module*, Publikation 1769-UM006.

Der Standardwert für die Ausgangsanordnung beträgt überall Null.

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Beschreibung | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Out15 | Out14 | Out13 | Out12 | Out11 | Out10 | Out09 | Out08 | Out07 | Out06 | Out05 | Out04 | Out03 | Out02 | Out01 | Out00 | OutputOnMask.0 -- OutputOnMask.15 | | |
| 1 | Out15 | Out14 | Out13 | Out12 | Out11 | Out10 | Out09 | Out08 | Out07 | Out06 | Out05 | Out04 | Out03 | Out02 | Out01 | Out00 | OutputOffMask.0 -- OutputOffMask.15 | | |
| 2 | R15 | R14 | R13 | R12 | R11 | R10 | R09 | R08 | R07 | R06 | R05 | R04 | R03 | R02 | R01 | R00 | RangeEn.0 -- RangeEn.15 | | |
| 3 | | | | | | | | | | | | | | | | | reserviert | | |
| 4 | | | | | | | | | RBF | | | | | | | | ResetBlownFuse | | |
| 5 | | | | | | | RPW | RREZ | Z Inh | Z Inv | D Inh | D Inv | RCU | RCO | SP | EN | Ctr0ControlBits | → | Ctr0En<br>Ctr0SoftPreset<br>Ctr0ResetCountOverflow<br>Ctr0ResetCountUnderflow<br>Ctr0DirectionInvert<br>Ctr0DirectionInhibit<br>Ctr0ZInvert<br>Ctr0ZInhibit<br>Ctr0ResetRisingEdgeZ<br>Ctr0ResetCtrPresetWarning |
| 6 | | | | | | | RPW | RREZ | Z Inh | Z Inv | D Inh | D Inv | RCU | RCO | SP | EN | Ctr1ControlBits | | |
| 7 | | | | | | | RPW | | | | | D Inv | RCU | RCO | SP | EN | Ctr2ControlBits | | |
| 8 | | | | | | | RPW | | | | | D Inv | RCU | RCO | SP | EN | Ctr3ControlBits | | |
| 9 | | | | | | | | | | | | | | | | | reserviert | | |
| 10<br>11 | Range12To15[0].HiLimOrDirWr | | | | | | | | | | | | | | | | Range12To15[0].HiLimOrDirWr | | |
| 12<br>13 | Range12To15[0].LowLimit | | | | | | | | | | | | | | | | Range12To15[0].LowLimit | | |
| 14 | Out15 | Out14 | Out13 | Out12 | Out11 | Out10 | Out09 | Out08 | Out07 | Out06 | Out05 | Out04 | Out03 | Out02 | Out01 | Out00 | Range12To15[0].OutputControl.0 ... .15 | | |
| 15 | | | | | | | | Inv | | | LDW | Typ | | | ToThisCtr | | Range12To15[0].Config Flags | → | Range12To15[0].ToThisCounter_0<br>Range12To15[0].ToThisCounter_1<br>Range12To15[0].Type<br>Range12To15[0].LoadDirectWrite<br>Range12To15[0].Invert |
| 16<br>17 | Range12To15[1].HiLimOrDirWr | | | | | | | | | | | | | | | | Range12To15[1].HiLimOrDirWr | | |
| 18<br>19 | Range12To15[1].LowLimit | | | | | | | | | | | | | | | | Range12To15[1].LowLimit | | |
| 20 | Out15 | Out14 | Out13 | Out12 | Out11 | Out10 | Out09 | Out08 | Out07 | Out06 | Out05 | Out04 | Out03 | Out02 | Out01 | Out00 | Range12To15[1].OutputControl.0 ... .15 | | |
| 21 | | | | | | | | Inv | | | LDW | Typ | | | ToThisCtr | | Range12To15[1].Config Flags | → | Range12To15[1].ToThisCounter_0<br>Range12To15[1].ToThisCounter_1<br>Range12To15[1].Type<br>Range12To15[1].LoadDirectWrite<br>Range12To15[1].Invert |
| 22<br>23 | Range12To15[2].HiLimOrDirWr | | | | | | | | | | | | | | | | Range12To15[2].HiLimOrDirWr | | |
| 24<br>25 | Range12To15[2].LowLimit | | | | | | | | | | | | | | | | Range12To15[2].LowLimit | | |
| 26 | Out15 | Out14 | Out13 | Out12 | Out11 | Out10 | Out09 | Out08 | Out07 | Out06 | Out05 | Out04 | Out03 | Out02 | Out01 | Out00 | Range12To15[2].OutputControl.0 ... .15 | | |
| 27 | | | | | | | | Inv | | | LDW | Typ | | | ToThisCtr | | Range12To15[2].Config Flags | → | Range12To15[2].ToThisCounter_0<br>Range12To15[2].ToThisCounter_1<br>Range12To15[2].Type<br>Range12To15[2].LoadDirectWrite<br>Range12To15[2].Invert |
| 28<br>29 | Range12To15[3].HiLimOrDirWr | | | | | | | | | | | | | | | | Range12To15[3].HiLimOrDirWr | | |
| 30<br>31 | Range12To15[3].LowLimit | | | | | | | | | | | | | | | | Range12To15[3].LowLimit | | |
| 32 | Out15 | Out14 | Out13 | Out12 | Out11 | Out10 | Out09 | Out08 | Out07 | Out06 | Out05 | Out04 | Out03 | Out02 | Out01 | Out00 | Range12To15[3].OutputControl.0 ... .15 | | |
| 33 | | | | | | | | Inv | | | LDW | Typ | | | ToThisCtr | | Range12To15[3].Config Flags | → | Range12To15[3].ToThisCounter_0<br>Range12To15[3].ToThisCounter_1<br>Range12To15[3].Type<br>Range12To15[3].LoadDirectWrite<br>Range12To15[3].Invert |

## *Eingangsanordnung für 1769-HSC-Hochgeschwindigkeitszähler-Modul*

Die Angaben in der nachfolgenden Tabelle stellen eine kurze Übersicht der Anordnung dar. Nähere Angaben finden Sie im Benutzerhandbuch *Compact™ High Speed Counter Module*, Publikation 1769-UM006.

Der Standardwert für die Eingangsanordnung ist überall Null.

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Beschreibung | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | Z1 | B1 | A1 | Z0 | B0 | A0 | InputStateA0 -- InputStateZ1 | |
| 1 | Out15 | Out14 | Out13 | Out12 | Out11 | Out10 | Out09 | Out08 | Out07 | Out06 | Out05 | Out04 | Out03 | Out02 | Out01 | Out00 | Readback.0 -- Readback.15 | |
| 2 | InvalidRangeLimit12…15 | | | | InvalidCtrAssignToRange12…15 | | | | GenErr | InvOut | MCfg | | Out0Overcurrent -- Out3… | | | | Status Flags → | InvalidRangeLimit12 ... 15<br>InvalidCtrAssignToRange12 ... 15<br>GenError<br>InvalidOutput<br>ModConfig<br>Out0Overcurrent0 ... 3 |
| 3 | R15 | R14 | R13 | R12 | R11 | R10 | R09 | R08 | R07 | R06 | R05 | R04 | R03 | R02 | R01 | R00 | RangeActive.0 -- RangeActive.15 | |
| 4<br>5 | Ctr[0].CurrentCount | | | | | | | | | | | | | | | | Ctr[0].CurrentCount | |
| 6<br>7 | Ctr[0].StoredCount | | | | | | | | | | | | | | | | Ctr[0].StoredCount | |
| 8<br>9 | Ctr[0].CurrentRate | | | | | | | | | | | | | | | | Ctr[0].CurrentRate | |
| 10<br>11 | Ctr[0].PulseInterval | | | | | | | | | | | | | | | | Ctr[0].PulseInterval | |
| 12 | | | | | | | | | | C0PW | RV | | IDW | REZ | CUdf | COvf | Ctr[0].StatusFlags → | Ctr[0].Overflow<br>Ctr[0].Underflow<br>Ctr[0].RisingEdgeZ<br>Ctr[0].InvalidDirectWrite<br>---<br>Ctr[0].RateValid<br>Ctr[0].PresetWarning |
| 13 | | | | | | | | | | | | | | | | | reserviert | |
| 14<br>15 | Ctr[1].CurrentCount | | | | | | | | | | | | | | | | Ctr[1].CurrentCount | |
| 16<br>17 | Ctr[1].StoredCount | | | | | | | | | | | | | | | | Ctr[1].StoredCount | |
| 18<br>19 | Ctr[1].CurrentRate | | | | | | | | | | | | | | | | Ctr[1].CurrentRate | |
| 20<br>21 | Ctr[1].PulseInterval | | | | | | | | | | | | | | | | Ctr[1].PulseInterval | |
| 22 | | | | | | | | | | C1PW | RV | IC | IDW | REZ | CUdf | COvf | Ctr[1].StatusFlags → | Ctr[1].Overflow<br>Ctr[1].Underflow<br>Ctr[1].RisingEdgeZ<br>Ctr[1].InvalidDirectWrite<br>Ctr[1].InvalidCounter<br>Ctr[1].RateValid<br>Ctr[1].PresetWarning |
| 23 | | | | | | | | | | | | | | | | | reserviert | |
| 24<br>25 | Ctr[2].CurrentCount | | | | | | | | | | | | | | | | Ctr[2].CurrentCount | |
| 26<br>27 | Ctr[2].CurrentRate | | | | | | | | | | | | | | | | Ctr[2].CurrentRate | |
| 28 | | | | | | | | | | C2PW | RV | IC | IDW | | CUdf | COvf | Ctr[2].StatusFlags → | Ctr[2].Overflow<br>Ctr[2].Underflow<br>---<br>Ctr[2].InvalidDirectWrite<br>Ctr[2].InvalidCounter<br>Ctr[2].RateValid<br>Ctr[2].PresetWarning |
| 29 | | | | | | | | | | | | | | | | | reserviert | |
| 30<br>31 | Ctr[3].CurrentCount | | | | | | | | | | | | | | | | Ctr[3].CurrentCount | |
| 32<br>33 | Ctr[3].CurrentRate | | | | | | | | | | | | | | | | Ctr[3].CurrentRate | |
| 34 | | | | | | | | | | C3PW | RV | IC | IDW | | CUdf | COvf | Ctr[3].StatusFlags → | Ctr[3].Overflow<br>Ctr[3].Underflow<br>---<br>Ctr[3].InvalidDirectWrite<br>Ctr[3].InvalidCounter<br>Ctr[3].RateValid<br>Ctr[3].PresetWarning |

### *Datenorganisation für 1769-SDN-DeviceNet-Scanner-Modul*

Der Scanner verwendet die Eingangs- und Ausgangsdatenbilder, um Daten-, Status- und Befehlsinformationen zwischen dem Scanner und der Steuerung zu übertragen. Im Folgenden wird die Grundstruktur dargestellt. Nähere Angaben finden Sie im Benutzerhandbuch *Compact I/O™ DeviceNet-Scannermodul 1769-SDN*, Publikation 1769-UM009.

#### Eingangsdatenbild

Das Eingangsdatenbild wird vom Scannermodul an die Steuerung übertragen.

| Wort | Beschreibung | Datentyp |
|---|---|---|
| 0 bis 63 | Statusstruktur | 64-Wort-Anordnung |
| 64 und 65 | Modulstatus-Register | 2 Wörter |
| 66 bis 245 | Eingangsdatenbild | 180-Wort-Anordnung |

#### Ausgangsdatenbild

Das Ausgangsdatenbild wird vom Scannermodul an die Steuerung übertragen.

| Wort | Beschreibung | Datentyp |
|---|---|---|
| 0 und 1 | Modul-Befehlsanordnung | 2-Wort-Anordnung |
| 2 bis 181 | Ausgangsdatenbild | 180-Wort-Anordnung |

Die nachfolgende Tabelle stellt eine Bit-Beschreibung für die Modul-Befehlsanordnung dar.

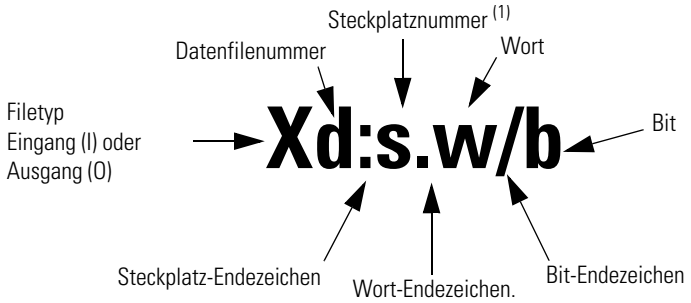
| Wort | Bit | Betriebsmodus |
|---|---|---|
| 0 | 0 | 1 = Betrieb, 0 = Leerlauf |
| | 1 | 1 = Fehler |
| | 2 | 1 = Netzwerk deaktivieren |
| | 3 | reserviert(1) |
| | 4 | 1 = zurücksetzen |
| | 5 bis 15 | reserviert(1) |
| 1 | 0 bis 15 | reserviert(1) |

(1) Verändern Sie die reservierten Bits NICHT. Dies könnte sonst Auswirkungen auf die weitere Kompatibilität haben.

# E/A-Adressierung

## Einzelheiten zur Adressierung

Nachfolgend sehen Sie das Prinzip der E/A-Adressierung sowie mehrere Beispiele.




(1) Die E/A auf der Steuerung (integrierte E/A) ist Steckplatz 0.
Die der Steuerung hinzugefügte E/A (Erweiterungs-E/A) beginnt mit Steckplatz 1.

| Format | Bedeutung | | |
|---|---|---|---|
| **Od:s.w/b**<br>**Id:s.w/b** | **x** | Filetyp | Eingang (E) und Ausgang (A) |
| | **d** | Datenfilenummer *(optional)* | 0 = Ausgang, 1 = Eingang |
| | **:** | Steckplatz-Endezeichen *(optional, nicht erforderlich für Datenfiles 2 bis 255)* | |
| | **s** | Steckplatznummer (dezimal) | Integrierte E/A: Steckplatz 0 |
| | | | Erweiterungs-E/A:<br>• Steckplätze 1 bis 6 für MicroLogix 1200 (eine Abbildung finden Sie auf Seite 1-3)<br>• Steckplätze 1 bis 16(1) für MicroLogix 1500 (eine Abbildung finden Sie auf Seite 1-9) |
| | **.** | Wort-Endezeichen. Nur erforderlich, wenn eine Wortnummer notwendig ist (siehe unten). | |
| | **w** | Wortnummer | Erforderlich zum Lesen/Schreiben von Wörtern und bei einer diskreten Bitnummer über 15.<br>Bereich: 0 bis 255 |
| | **/** | Bit-Endezeichen | |
| | **b** | Bitnummer | 0 bis 15 |

(1) Steckplätze 1 bis 8 für Basisgeräte der Serie A.

## Beispiele für die Adressierung

| Adressierungsebene | Beispieladresse(1) | Steckplatz | Wort | Bit |
|---|---|---|---|---|
| **Bitadressierung** | O:0/4(2) | Ausgangssteckplatz 0 (integrierte E/A) | Wort 0 | Ausgangs-Bit 4 |
| | O:2/7(2) | Ausgangssteckplatz 2 (Erweiterungs-E/A) | Wort 0 | Ausgangs-Bit 7 |
| | I:1/4(2) | Eingangssteckplatz 1 (Erweiterungs-E/A) | Wort 0 | Eingangs-Bit 4 |
| | I:0/15(2) | Eingangssteckplatz 0 (integrierte E/A) | Wort 0 | Eingangs-Bit 15 |
| **Wortadressierung** | O:1.0 | Eingangssteckplatz 1 (Erweiterungs-E/A) | Wort 0 | |
| | I:7.3 | Eingangssteckplatz 7 (Erweiterungs-E/A) | Wort 3 | |
| | I:3.1 | Eingangssteckplatz 3 (Erweiterungs-E/A) | Wort 1 | |

(1) Die optionale Datenfilenummer wird in diesen Beispielen nicht verwendet.

(2) Wortendezeichen und -anzahl werden nicht aufgeführt. Deshalb gilt die Adresse für Wort 0.

## E/A-Forcen

Beim Forcen von E/A wird der aktuelle Status der E/A durch den Benutzer überschrieben.

### Forcen des Eingangs

Beim Forcen eines Eingangs wird der Wert im Eingangsdatenfile auf einen benutzerdefinierten Status gesetzt. Bei diskreten Eingängen kann ein Eingang auf diese Weise aktiviert oder deaktivert werden. Nach Forcen eines Eingangs gibt der betreffende Eingang nicht mehr den Zustand des physischen Eingangs oder der Eingangs-LED wieder. Bei integrierten Eingängen reagiert die Steuerung, als ob die physische Eingangsklemme forciert wurde.

**HINWEIS** Das Forcen eines Eingangs hat keine Auswirkungen auf das Eingangsgerät, das an die Steuerung angeschlossen ist.

### Forcen des Ausgangs

Beim Forcen eines Ausgangs übergeht die Steuerung den Status des Steuerprogramms und setzt den Ausgang auf den benutzerdefinierten Status. Bei diskreten Ausgängen kann der Ausgang auf diese Weise aktiviert oder deaktiviert werden. Der Wert in der Ausgangsdatei bleibt davon unberührt; er behält den Status bei, der durch die Logik des Steuerprogramms ermittelt wurde. Der Status des physischen Ausgangs und der Ausgangs-LED wird allerdings auf den neuen Status gesetzt.

**HINWEIS** Beim Forcen eines Ausgangs, der durch eine ausführende PTO- oder PWM-Funktion gesteuert wird, wird ein Befehlsfehler generiert.

## Eingangsfilter

Benutzer von MicroLogix 1200- und MicroLogix 1500-Steuerungen können DC-Eingangsgruppen für den Hochgeschwindigkeits- oder den Standardbetrieb einrichten. Dabei kann auch die Antwortzeit für jede Eingangsgruppe konfiguriert werden. Ein konfigurierbarer Filter legt fest, wie lange das Eingangssignal auf „ein“ oder „aus“ gesetzt sein muss, bis das Signal von der Steuerung erkannt wird. Je höher der Wert, um so mehr Zeit vergeht, bis der Eingangszustand durch die Steuerung erkannt wird. Höhere Werte, die eine umfassendere Filterung bewirken, werden vor allem in Umgebungen mit starken elektrischen Störungen verwendet. Niedrigere Werte führen zu einer niedrigeren Filterwirkung und werden zur Erkennung schneller oder schmaler Impulse eingesetzt. Bei Hochgeschwindigkeitszählern, Impulsspeicher-Eingängen und Eingangs-Interrupts wird in der Regel ein niedriger Filterwert festgelegt.

Der Eingangsfilter wird über die Programmiersoftware RSLogix 500 konfiguriert. Gehen Sie dabei wie folgt vor:

1. Öffnen Sie den Ordner „Controller“ (Steuerung).
2. Öffnen Sie den Ordner „I/O Configuration“ (E/A-Konfiguration).
3. Öffnen Sie den Steckplatz 0 (Steuerung).
4. Wählen Sie die Markierung „Embedded I/O Configuration“ (Integrierte E/A-Konfiguration).

Die Eingangsgruppen sind voreingestellt. Wählen Sie die gewünschte Filterzeit für jede Eingangsgruppe aus. Sie können für jede der Eingangsgruppen einen anderen Eingangsfilterwert festlegen:

| Steuerung | MicroLogix 1200 | MicroLogix 1500 |
|---|---|---|
| **Eingangsgruppen** | • 0 und 1<br>• 2 und 3<br>• 4 und darüber | • 0 und 1<br>• 2 und 3<br>• 4 und 5<br>• 6 und 7<br>• 8 und darüber |

Die Mindest- und Maximalantwortzeiten zu jedem Eingangsfilterwert finden Sie im Benutzerhandbuch zu Ihrer Steuerung.

## Impulsspeicher-Eingänge

MicroLogix 1200- und MicroLogix 1500-Steuerungen bieten die Möglichkeit der getrennten Konfiguration der einzelnen Eingänge als Impulsspeicher-Eingänge. Ein Impulsspeicher-Eingang ist ein Eingang, der einen sehr schnellen Impuls erfasst und für die Dauer einer Steuerungsabfrage speichert. Die Impulsdauer, die erfasst werden kann, hängt von dem Eingangsfilter ab, der für den betreffenden Eingang ausgewählt wurde.

Folgende Eingänge können als Impulsspeicher-Eingänge konfiguriert werden:

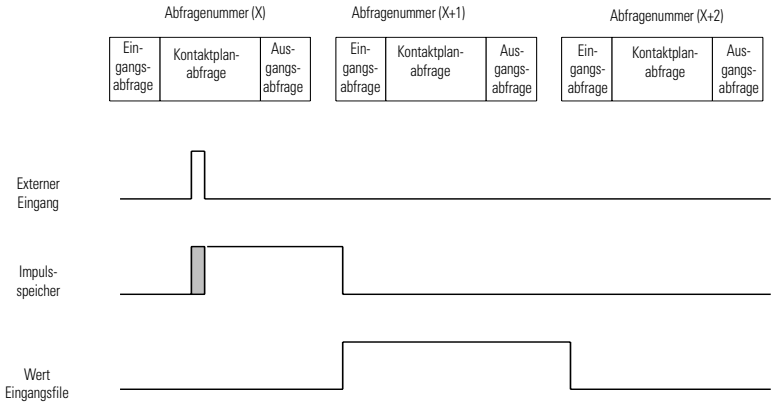
| Steuerung | MicroLogix 1200 | MicroLogix 1500 |
|---|---|---|
| **DC-Eingänge** | 0 bis 3 | 0 bis 7 |

Diese Funktion wird über die Programmiersoftware RSLogix 500 aktiviert. Gehen Sie dabei nach dem Öffnen eines Projekts wie folgt vor:

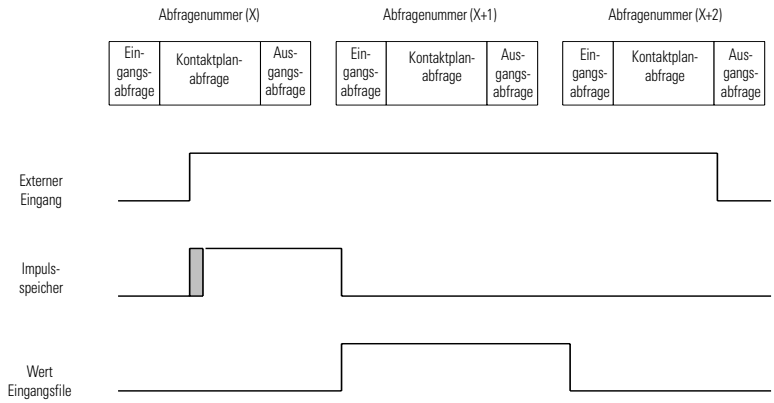
1. Öffnen Sie den Ordner „Controller“ (Steuerung).
2. Öffnen Sie den Ordner „I/O Configuration“ (E/A-Konfiguration).
3. Öffnen Sie den Steckplatz 0 (Steuerung).
4. Wählen Sie die Markierung „Embedded I/O Configuration“ (Integrierte E/A-Konfiguration).
5. Wählen Sie die Masken-Bits der Eingänge aus, die als Impulsspeicher-Eingänge verwendet werden sollen.
6. Wählen Sie den Status der Impulsspeicher-Eingänge aus. In Abhängigkeit von der Konfiguration, die in der Programmier- software ausgewählt wurde, erkennt die Steuerung „Einschalt- impulse“ (Anstiegsflanke) oder „Ausschaltimpulse“ (abfallende Flanke).

Die folgenden Informationen gelten für eine Steuerung bei Erfassung eines Einschaltimpulses. Bei Erkennung eines externen „Einschalt- signals“ speichert die Steuerung dieses Ereignis. In der Regel wird der Eingangsdatenpunkt bei der nächsten Eingangsabfrage nach diesem Ereignis auf „ein“ gesetzt; dieser Zustand bleibt während der nächsten Steuerungsabfrage bestehen. Bei der darauf folgenden Eingangs- abfrage wird der Wert wieder auf „aus“ gesetzt. Die folgenden Abbildungen illustrieren diesen Vorgang.

## Verhalten bei Anstiegsflanke - Beispiel 1




## Verhalten bei Anstiegsflanke - Beispiel 2



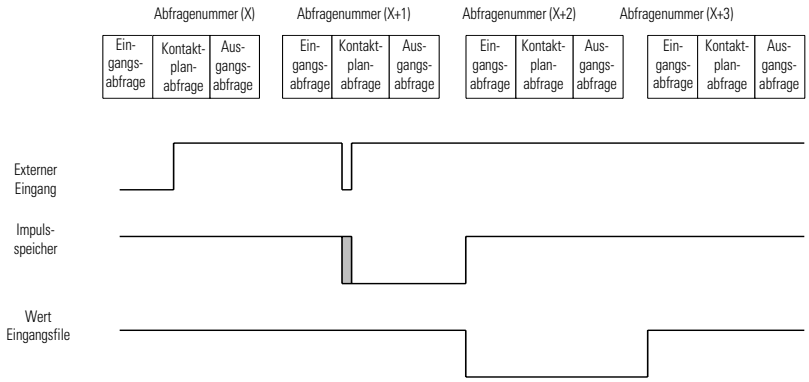

**HINWEIS** Der graue Bereich der Kurve „Impulsspeicher“ stellt die Eingangsfilterverzögerung dar.

**WICHTIG** Der Wert der Eingangsdatei stellt nicht den externen Eingang dar, wenn der Eingang als Impulsspeicher konfiguriert wurde. Bei der Konfiguration zur Erkennung der Anstiegsflanke ist der Wert des Eingangsfiles in der Regel „aus“ („ein“ während einer Abfrage nach Erkennung eines Impulses mit Anstiegsflanke.)

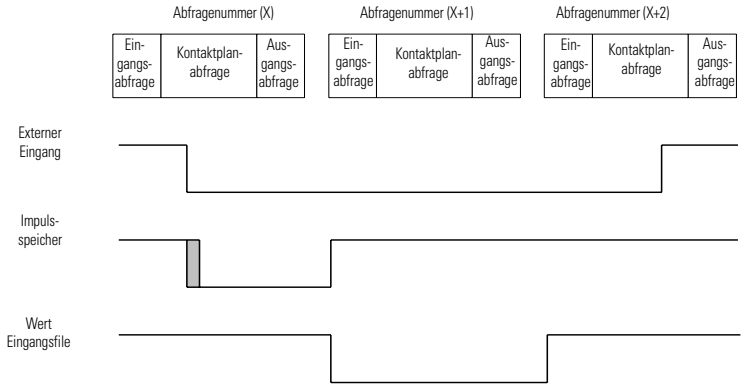
Die oben aufgeführten Beispiele verdeutlichen das Verhalten der Anstiegsflanke. Das Verhalten für abfallende Flanken ist bis auf folgende Ausnahmen mit diesem Verhalten identisch:

- Die Erkennung erfolgt an der abfallende Flanke des externen Eingangs.
- Die Eingangsdaten sind in der Regel „ein“ (1) und werden für die Dauer einer Abfrage auf „aus“ (0) gesetzt.

## Verhalten bei abfallender Flanke - Beispiel 1




## Verhalten bei abfallender Flanke - Beispiel 2




**HINWEIS** Der graue Bereich der Kurve „Impulsspeicher“ stellt die Eingangsfilterverzögerung dar.

**WICHTIG** Der Wert der Eingangsdatei stellt nicht den externen Eingang dar, wenn der Eingang als Impulsspeicher konfiguriert wurde. Bei der Konfiguration zur Erkennung der abfallenden Flanke ist der Wert der Eingangsdatei in der Regel „ein“ („aus“ während einer Abfrage nach Erkennung eines Impulses mit abfallender Flanke.)

## Konfiguration der Erweiterungs-E/A mit RSLogix 500

Die Erweiterungs-E/A müssen für den Einsatz mit der Steuerung konfiguriert werden. Die Konfiguration der Erweiterungs-E/A kann entweder manuell oder automatisch erfolgen. Verwenden von RSLogix 500:

1. Öffnen Sie den Ordner „Controller“ (Steuerung).
2. Öffnen Sie den Ordner „I/O Configuration“ (E/A-Konfiguration).
3. Ziehen Sie zur manuellen Konfiguration das Compact-E/A-Modul auf den Steckplatz.

   Für die automatische Konfiguration muss die Steuerung am Computer angeschlossen sein (entweder direkt oder über ein Netzwerk). Klicken Sie Im E/A-Konfigurationsbildschirm auf die Schaltfläche „Read I/O Config“ (E/A-Konfiguration lesen). RSLogix 500 liest die aktuelle Konfiguration der Steuerungs-E/A ein.

Einige E/A-Module unterstützen oder erfordern die Konfiguration. Zum Konfigurieren eines speziellen Moduls klicken Sie doppelt auf das gewünschte Modul. Anschließend wird ein spezieller E/A- Konfigurationsbildschirm für dieses Modul angezeigt.

# Speicher der Steuerung und Filetypen

In diesem Kapitel werden der Speicher der Steuerung und die Filetypen beschrieben, die von MicroLogix 1200- und MicroLogix 1500-Steuerungen verwendet werden. Dabei werden folgende Themen erläutert:


- „Speicher der Steuerung“ auf Seite 2-2
- „Datenfiles“ auf Seite 2-7
- „Datenfiles beim Herunterladen schützen“ auf Seite 2-8
- „Statischer Fileschutz“ auf Seite 2-10
- „Kennwortschutz“ auf Seite 2-11
- „Speicher der Steuerung löschen“ auf Seite 2-12
- „Zukünftigen Zugriff zulassen (OEM-Sperre)“ auf Seite 2-13

# Speicher der Steuerung

## Filestruktur

Der MicroLogix 1200- und 1500-Anwenderspeicher besteht aus Datenfiles, Funktionsfiles und Programmfiles (sowie B-RAM-Files für den MicroLogix 1500 1764-LRP-Prozessor). Funktionsfiles stehen ausschließlich in den MicroLogix 1200- und 1500-Steuerungen
zur Verfügung. Sie sind nicht in den MicroLogix 1000- oder SLC-Steuerungen verfügbar.

**HINWEIS** Die nachfolgend aufgeführten Filetypen für die Datenfiles 3 bis 7 sind die Standardfile- typen für die Filenummern. Sie können nicht geändert werden. Die Datenfiles 9 bis 255 können Ihrem Programm hinzugefügt werden. Sie dienen als Bit-, Zeitwerk-, Zähler-, Steuerungs-, Ganzzahlen-, Zeichenketten-, Doppelwort-, Nachrichten- oder PID-Files.

| Datenfiles | | Funktionsfiles | | Programmfiles | | Spezielle Files(1) | |
|---|---|---|---|---|---|---|---|
| 0 | Ausgangsfile | HSC | Hochgeschwindigkeitszähler | 0 | Systemfile 0 | 0 | Datenprotokoll-Warteschleife 0 |
| 1 | Eingangsfile | PTO(2) | Frequenzausgang | 1 | Systemfile 1 | 1 | Datenprotokoll-Warteschleife 1 |
| 2 | Statusfile | PWM(2) | Pulsweitenmodulation | 2 | Programmfile 2 | 2 bis 255 | Datenprotokoll-Warteschleifen 2 bis 255 |
| 3 | Bitfile | STI | Wählbarer zeitgesteuerter Interrupt | 3 bis 255 | Programmfiles 3 bis 255 | 0 | Rezeptfile 0 |
| 4 | Zeitwerkfile | EII | Ereigniseingangs-Interrupt | | | 1 | Rezeptfile 1 |
| 5 | Zählerfile | RTC | Echtzeituhr | | | 2 bis 255 | Rezeptfiles 2 bis 255 |
| 6 | Steuerungsfile | TPI | Einstell-potentiometerdaten | | | | |
| 7 | Ganzzahl-File | MMI | Speichermoduldaten | | | | |
| 8 | Fließkommafile | DAT(3) | Datenzugriffsmodul | | | | |
| 9 bis 255 | (B) Bit<br>(T) Zeitwerk<br>(C) Zähler<br>(R) Steuerung<br>(N) Ganzzahl<br>(F) Fließkomma(4)<br>(ST) Zeichenkette(5)<br>(L) Doppelwort<br>(MG) Meldung<br>(PD) PID<br>(PLS) Programmierbarer Endschalter(4) | BHI | Basis-Hardware-Information | | | | |
| | | CS | Kommunikationsstatus | | | | |
| | | IOS | E/A-Status | | | | |
| | | DLS(1) | Datenprotokollstatus | | | | |

(1) Spezielle Files zur Datenprotokollierung werden nur vom MicroLogix 1500 1764-LRP-Prozessor verwendet. Spezielle Files für Rezepte werden nur von MicroLogix 1500-Prozessoren der Serie C verwendet.

(2) Die PTO- und PWM-Files werden nur in MicroLogix 1200- und 1500 BXB-Geräten verwendet.

(3) Die DAT-Files werden nur in MicroLogix 1500-Steuerungen verwendet.

(4) Die Fließkomma- und programmierbaren Endschalter-Files sind in MicroLogix 1200- und 1500-Steuerungen der Serie C verfügbar.

(5) Der Zeichenkettenfile steht in MicroLogix 1200-Steuerungen und MicroLogix 1500 1764-LSP-Prozessoren der Serie B (und später) und 1764-LRP-Prozessoren zur Verfügung.

## Anwenderspeicher

Der Anwenderspeicher ist die Speichermenge, die einem Anwender zum Speichern der Kontaktplanlogik, der Datentafelfiles, der E/A-Konfiguration, usw., in der Steuerung zur Verfügung steht.

Zu den Anwenderdatenfiles gehören die Systemstatusfiles, E/A-Imagefiles sowie alle anderen durch Anwender erstellbaren Datenfiles (Bit, Zeitwerk, Zähler, Steuerung, Ganzzahl, Zeichenkette, Doppelwort, MSG und PID).

Ein Wort ist definiert als eine Speichereinheit, die in der Steuerung belegt wird. Die dem Anwender zur Verfügung stehende Speicher- menge für Daten- und Programmfiles wird in Anwenderworten gemessen. Die Speicherbelegung wird wie folgt zugeordnet:

- Bei Datenfiles entspricht ein Wort 16 Speicher-Bits. Beispiel:
  - 1 ganzzahliges Datenfile-Element = 1 Anwenderwort
  - 1 Doppelwortfile-Element = 2 Anwenderworte
  - 1 Zeitwertdatenfile-Element = 3 Anwenderworte

**HINWEIS** Jedes Ein- und Ausgangsdatenelement ver- braucht 3 Anwenderworte aufgrund des mit dem E/A-Forcen verbundenem Overhead.

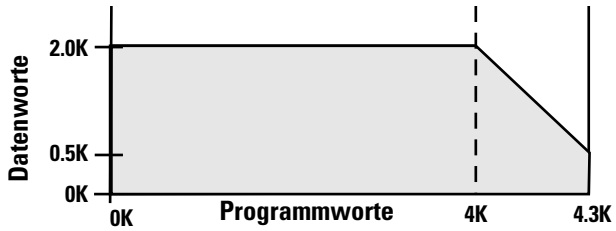
- Bei Programmfiles entspricht ein Wort einem Kontaktplanbefehl mit einem Operanden. Beispiel(1):
  - 1 XIC-Befehl mit einem Operanden belegt 1 Anwenderwort
  - 1 EQU-Befehl mit 2 Operanden belegt 2 Anwenderworte
  - 1 ADD-Befehl mit 3 Operanden belegt 3 Anwenderworte
- *Funktionsfiles* belegen keinen Anwenderspeicherplatz.

**HINWEIS** Obwohl die Steuerung bis zu 256 Elemente pro File unterstützt, ist es unter Umständen aufgrund der Größe des Anwenderspeichers in der Steuerung nicht möglich, einen File mit dieser Anzahl an Elementen zu erstellen.

(1) Hierbei handelt es sich um Näherungswerte. Informationen zur tatsächlichen Speicherbelegung entnehmen Sie bitte den Tabellen in Anhang A und B dieses Handbuchs.

### *Anwenderspeicher bei MicroLogix 1200*

Die MicroLogix 1200-Steuerung unterstützt 6 KB Speicher. Der Speicher kann für Programm- und Datenfiles verwendet werden. Die maximale Datenspeichermenge für Wörter beträgt 2 KB (siehe unten).

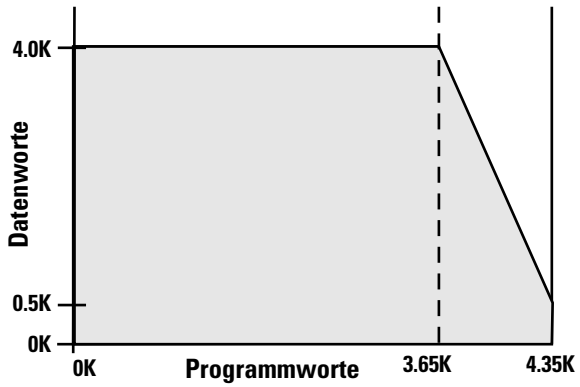



Informationen zur Speicherbelegung sowie spezielle Anweisungen finden Sie im Abschnitt „Speicherbelegung und Ausführungszeiten der MicroLogix 1200-Programmierbefehle“ auf Seite A-1.

### *Anwenderspeicher bei MicroLogix 1500*

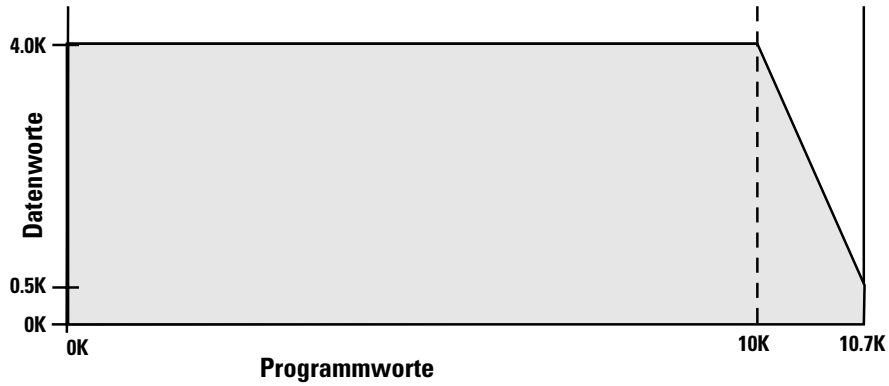
**MicroLogix 1500, 1764-LSP-Prozessor**

Der 1764-LSP-Prozessor unterstützt über 7 KB Speicher. Der Speicher kann für Programm- und Datenfiles verwendet werden. Die maximale Datenspeichermenge für Wörter beträgt 4 KB (siehe unten).



**MicroLogix 1500, 1764-LRP-Prozessor**

Der 1764-LRP-Prozessor unterstützt 14 KB Speicher. Der Speicher kann für Programm- und Datenfiles verwendet werden. Die maximale Datenspeichermenge für Wörter beträgt 4 KB (siehe unten).

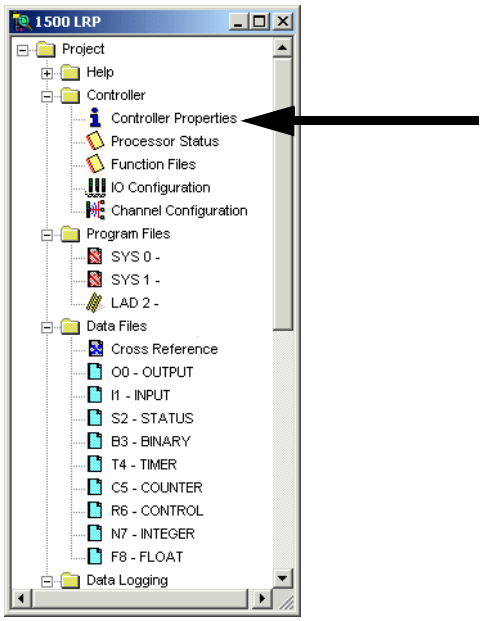



**WICHTIG** Für die MicroLogix 1500-Steuerung liegt die maximale Filegröße eines einzelnen Kontaktplanfiles bei 64.000 Worten. Sie können den gesamten Programmierungsspeicher nutzen, wenn Sie mehrere Kontaktplanfiles über Unterprogramme verwenden.

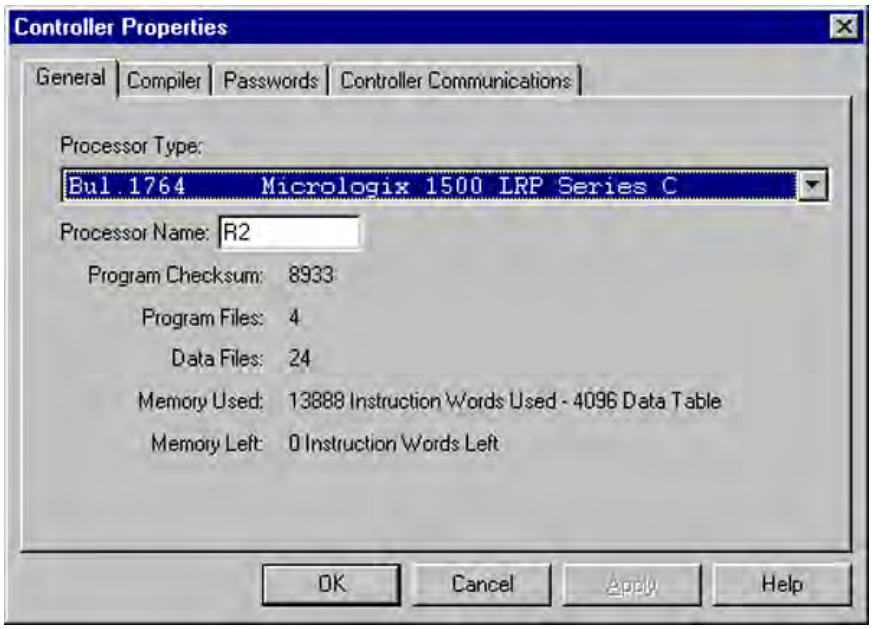
Der 1764-LRP-Prozessor unterstützt auch 48 KB batteriegestützten Speicher für Datenprotokollier- oder Rezept-Vorgänge. Informationen zur Datenprotokollierung und zu Rezepten finden Sie in Kapitel 22. Informationen zur Speicherbelegung sowie spezielle Anweisungen finden Sie im Abschnitt „MicroLogix 1500 – Speicherbelegung und Befehlsausführungszeit“ auf Seite B-1.

## Steuerungsspeicherbelegung anzeigen

1. Markieren und öffnen Sie *Controller Properties* (Eigenschaften der Steuerung).




2. Die Werte für *Memory Used* (Verwendeter Speicher) und *Memory Left* (Freier Speicher) werden dann im Fenster *Controller Properties* (Eigenschaften der Steuerung) angezeigt.



## Datenfiles

Datenfiles enthalten numerische Informationen, einschließlich E/A, Status und andere Daten, die zu den Befehlen gehören, die in Kontaktplan-Unterprogrammen verwendet werden. Die Datenfile- typen sind:

| Filename | File-kennung | File-nummer(1) | Worte pro Element | Filebeschreibung |
|---|---|---|---|---|
| Ausgangsfile | O | 0 | 1 | In dem Ausgangsfile werden die Werte gespeichert, die während der Ausgangsabfrage in die physischen Ausgänge geschrieben werden. |
| Eingangsfile | I | 1 | 1 | In dem Eingangsfile werden die Werte gespeichert, die während der Eingangsabfrage aus den physischen Eingängen gelesen werden. |
| Statusfile | S | 2 | 1 | Der Inhalt des Statusfiles hängt von den Funktionen ab, die den Statusfile verwenden. Eine ausführliche Beschreibung finden Sie im Abschnitt „Systemstatusfile“ auf Seite C-1. |
| Bitfile | B | **3**, 9 bis 255 | 1 | Der Bitfile ist ein Allzweckfile, der in der Regel bei der Bitlogik eingesetzt wird. |
| Zeitwerkfile | T | **4**, 9 bis 255 | 3 | In dem Zeitwerkfile werden Zeitmessungsinformationen gespeichert, die von Zeitmessungsbefehlen der Kontaktplanlogik verwendet werden. Anweisungen finden Sie im Abschnitt „Zeitwerk- und Zählerbefehle“ auf Seite 8-1. |
| Zählerfile | C | **5**, 9 bis 255 | 3 | In dem Zählerfile werden Zählerinformationen gespeichert, die von Zählerbefehlen der Kontaktplanlogik verwendet werden. Anweisungen finden Sie im Abschnitt „Zeitwerk- und Zählerbefehle“ auf Seite 8-1. |
| Steuerungsfile | R | **6**, 9 bis 255 | 3 | Der Steuerdatenfile wird zur Speicherung der Längen- und Positions-angaben für verschiedene Befehle der Kontaktplanlogik verwendet. Weitere Informationen hierzu finden Sie unter „Steuerdatenfile“ auf Seite 20-6. |
| Ganzzahlfile | N | **7**, 9 bis 255 | 1 | Der Ganzzahl-File ist ein Allzweckfile, der aus 16-Bit-Datenwörtern (Ganzzahl mit Vorzeichen) besteht. |
| Fließkommafile | F | **8**, 9 bis 255 | 1 | Der Fließkommafile ist ein allgemeiner File, der aus 32-Bit-IEEE-754-Fließ-kommadatenelementen besteht. Weitere Informationen finden Sie unter „Fließkomma-Datenfile (F) verwenden“ auf Seite 10-4. |
| Zeichenketten-File | ST | 9 bis 255 | 42 | In dem Zeichenketten-File werden ASCII-Zeichen gespeichert. Weitere Informationen finden Sie unter „ST-Datenfile (String; Zeichenkette)“ auf Seite 20-5. |
| Doppelwortfile | L | 9 bis 255 | 2 | Der Doppelwort-File ist ein Allzweckfile, der aus 32-Bit-Datenwörtern (Ganzzahl mit Vorzeichen) besteht. |
| Nachrichtenfile | MG | 9 bis 255 | 25 | Der Nachrichtenfile gehört zu dem MSG-Befehl. Informationen zum MSG-Befehl finden Sie im Abschnitt „Kommunikationsbefehle“ auf Seite 21-1. |
| Program-mierbarer Endschalter-File | PLS | 9 bis 255 | 6 | Mit dem programmierbaren Endschalter (PLS)-File können Sie den Hochgeschwindigkeitszähler so konfigurieren, dass er als ein PLS oder als Nockendrehschalter arbeitet. Weitere Informationen finden Sie unter „Programmierbarer Endschalter-File (PLS)“ auf Seite 5-28. |
| PID-File | PD | 9 bis 255 | 23 | Der PID-File gehört zu dem PID-Befehl. Weitere Informationen hierzu finden Sie im Abschnitt „Prozesssteuerungsbefehl“ auf Seite 19-1. |

(1) Die **FETTGEDRUCKTE** Filenummer entspricht der Standardeinstellung. Weitere Datenfiles dieses Typs können mit Hilfe der verbleibenden Nummern konfiguriert werden.

# Datenfiles beim Herunterladen schützen

## Herunterladeschutz für Datenfiles

Sobald ein Anwenderprogramm in die Steuerung geladen wurde, ist es unter Umständen erforderlich, die Kontaktplanlogik zu aktualisieren und anschließend auf die Steuerung herunterzuladen, ohne dabei die von dem Anwender konfigurierten Variablen in einem oder mehreren Datenfiles in der Steuerung zu zerstören. Diese Situation kann sich ergeben, wenn eine Anwendung aktualisiert werden muss, während die Daten, die für die Installation relevant sind, unversehrt bleiben müssen.

Diese Funktion wird als *Schutzfunktion für Datenfiles beim Herunterladen* bezeichnet. Die Schutzfunktion wird in folgenden Fällen wirksam:

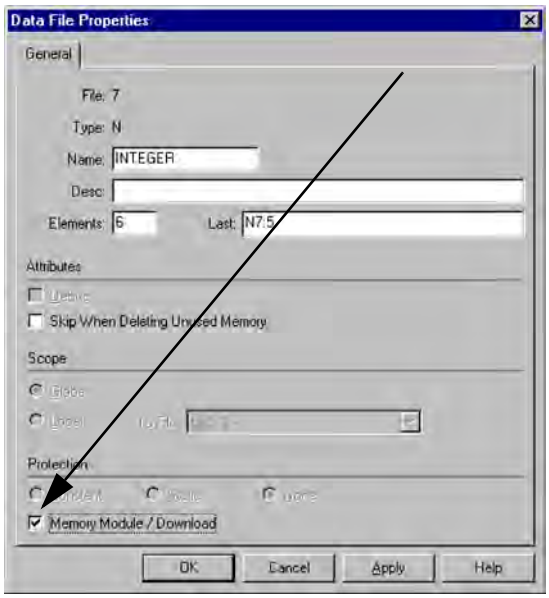
- Ein Anwenderprogramm wird über die Programmiersoftware heruntergeladen
- Ein Anwenderprogramm wird von einem Speichermodul heruntergeladen

### *Festlegen der Schutzfunktion für heruntergeladene Files*

Die Schutzfunktion für heruntergeladene Files kann für folgende Filetypen aktiviert werden:

- Ausgang (O)
- Eingang (I)
- Binär (B)
- Zeitwerk (T)
- Zähler (C)
- Steuerung (R)
- Ganzzahl (N)
- Fließkomma (F)
- Zeichenkette (ST)
- Doppelwort (L)
- Proportional Integral Derivative (PID)
- Nachricht (MG)
- Programmierbarer Endschalter (PLS)

**HINWEIS** Die Daten eines Statusfiles können nicht geschützt werden.




Die Funktion zum Schutz heruntergeladener Datenfiles können Sie über die Programmiersoftware RSLogix 500 aktivieren. Wählen Sie für jeden Datenfile, der geschützt werden soll, in dem Fenster „Data File Properties“ (Eigenschaften Datenfile) den Eintrag „Memory Module/Download“ (Speichermodul/Herunterladen) in der Schutzliste aus (siehe unten). Dieses Fenster wird angezeigt, wenn Sie den gewünschten Datenfile mit der rechten Maustaste anklicken.

### *Bedingungen für Fileschutz bei Anwenderprogrammübertragung*

Die Schutzfunktion für heruntergeladene Datenfiles ist wirksam, wenn während des Herunterladens eines Anwenderprogramms oder der Übertragung eines Speichermoduls zur Steuerung fol- gende Bedingungen erfüllt sind:

- Die Steuerung enthält geschützte Datenfiles.
- Das heruntergeladene Programm weist dieselbe Anzahl geschützter Datenfiles auf wie das aktuell in der Steuerung geladene Programm.
- Alle geschützten Datenfilenummern, -typen und -größen (Anzahl der Elemente), die sich derzeit in der Steuerung befinden, stimmen exakt mit dem in die Steuerung heruntergeladenen Programm überein.

Werden alle diese Bedingungen erfüllt, überschreibt die Steuerung keine der Datenfiles in der Steuerung, die als geschützt konfiguriert sind.

Werden nicht alle diese Bedingungen erfüllt, wird das gesamte Anwenderprogramm zur Steuerung übertragen. Enthält das Programm in der Steuerung außerdem geschützte Files, wird der Anzeiger für den Datenschutzverlust (S:36/10) gesetzt, um darauf hinzuweisen, dass geschützte Daten verloren gegangen sind. Beispiel: Ein Steuerpro- gramm mit geschützten Dateien wird auf die Steuerung übertragen. Das Originalprogramm enthält keine geschützten Dateien, oder die Dateien sind nicht identisch bezüglich Anzahl, Typ usw. Daraufhin wird der Anzeiger für den Datenschutzverlust (S:36/10) aktiviert. Der Anzeiger für den Datenschutzverlust weist darauf hin, dass für die geschützten Files innerhalb der Steuerung Werte heruntergeladen wurden und das Anwenderprogramm eventuell erneut konfiguriert werden muss.

**HINWEIS** Das Status-Bit für den Datenschutzverlust wird nicht durch die Steuerung gelöscht. Dieses Bit muss durch den Anwender gelöscht werden.

## Statischer Fileschutz

Wenn ein Datenfile über einen statischen Fileschutz verfügt, können die in ihm enthaltenen Werte nicht über die Kommunikations- schnittstelle verändert werden (Ausnahme ist das Herunterladen eines Programms zur Steuerung).

### Verwenden des statischen Fileschutzes mit Datenfileschutz beim Herunterladen

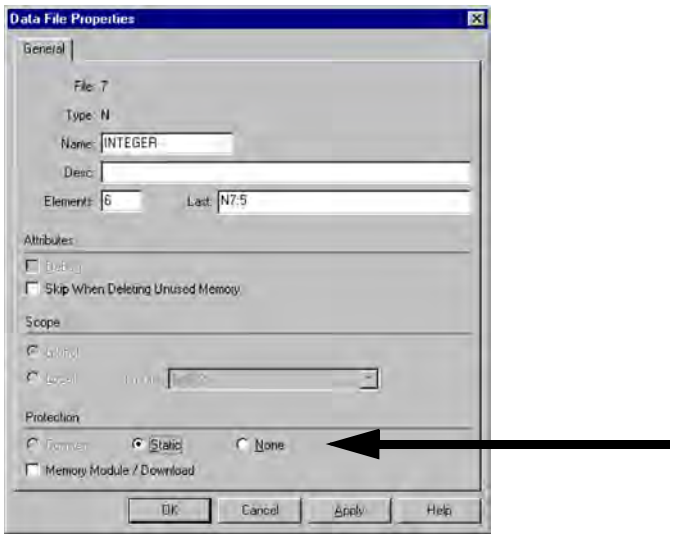
Statischer Fileschutz und Datenfileschutz beim Herunterladen können in Kombination mit einer beliebigen MicroLogix 1200-Steuerung der Serie B und höher sowie einem MicroLogix 1500-Prozessor der Serie B und höher verwendet werden.

### Festlegen der statischen Fileschutzfunktion

Der statische Fileschutz kann für folgende Filetypen aktiviert werden:

- Ausgang (O)
- Eingang (I)
- Status (S)
- Binär (B)
- Zeitwerk (T)
- Zähler (C)
- Steuerung (R)
- Ganzzahl (N)
- Fließkomma (F)
- Zeichenkette (ST)
- Doppelwort (L)
- Proportional Integral Derivative (PID)
- Meldung (MG)
- Programmierbarer Endschalter (PLS)

Die statische Fileschutzfunktion können Sie über die RSLogix500-Programmiersoftware aktivieren. Wählen Sie im Bildschirm „Data File Properties“ (Eigenschaften Datenfile) die statische Schutzfunktion für jeden zu schützenden Datenfile aus (siehe die folgende Abbildung). Dieses Fenster wird angezeigt, wenn Sie den gewünschten Datenfile mit der rechten Maustaste anklicken.

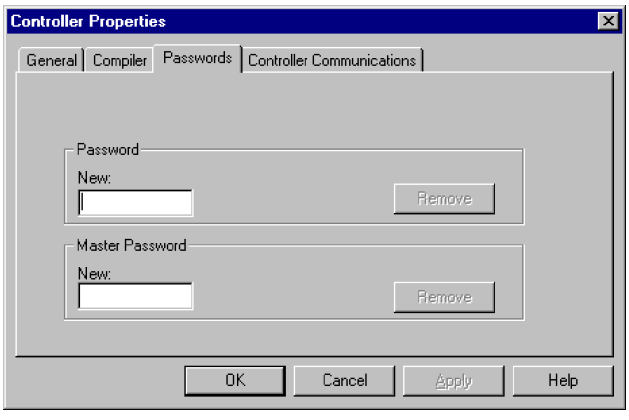



## Kennwortschutz

MicroLogix-Steuerungen verfügen über ein integriertes Sicherheits- system mit numerischen Kennworten. Diese Kennworte bestehen aus bis zu 10 Ziffern (0-9). Für jedes Steuerungsprogramm können zwei Kennworte festgelegt werden, das Kennwort und das Master-Kennwort.

Mit Kennworten wird der Zugriff auf die Steuerung eingeschränkt. Mit dem Master-Kennwort können Sie sich über den normalen Kenn- wortschutz hinwegsetzen. Der Kennwortschutz beruht auf dem folgenden System: Für alle Steuerungen eines Projekts werden unter- schiedliche Kennworte, aber dasselbe Master-Kennwort festgelegt, um einen einfachen Zugriff auf alle Steuerungen für Verwaltungs- und Wartungszwecke zu ermöglichen.

Sie können ein Kennwort über das Dialogfenster „Controller Properties“ (Eigenschaften der Steuerung) festlegen, ändern oder löschen. Die Verwendung von Kennworten ist nicht obligatorisch; ein Master-Kennwort wird jedoch nur dann vom System erkannt, wenn auch ein normales Kennwort verwendet wird.



**HINWEIS** Bei Verlust oder Vergessen eines Kennworts gibt es keine Möglichkeit, das Kennwort zu umgehen und das Programm wiederherzustellen. In diesem Fall muss der Speicher der Steuerung gelöscht werden.

Wenn in dem Speichermodul-Anwenderprogramm die Funktion „Load Always“ (Immer laden) aktiviert ist und für die Steuerung „Anwender-programm“ ein Kennwort festgelegt wurde, vergleicht die Steuerung die Kennworte, bevor das Anwenderprogramm von dem Speicher- modul auf die Steuerung übertragen wird. Bei fehlender Überein- stimmung der Kennworte werden das Anwenderprogramm nicht übertragen und das Bit für die fehlende Programmübereinstimmung gesetzt (S:5/9).

## Speicher der Steuerung löschen

Wenn Sie keinen Zugriff auf die Steuerung haben, weil Sie nicht über das Kennwort verfügen, können Sie den Speicher der Steuerung löschen und ein neues Anwenderprogramm herunterladen.

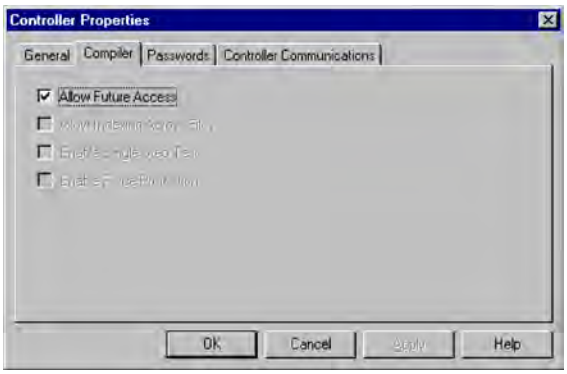
Sie können den Speicher löschen und eine Online-Verbindung für die Steuerung herstellen, wenn eine Aufforderung der Programmier- software zur Eingabe eines System- oder Master-Kennworts angezeigt wird. Gehen Sie dabei wie folgt vor:

1. Geben Sie die Zahl 65257636 ein (dies entspricht der Zeichenfolge MLCLRMEM – MicroLogix Clear Memory – auf einer Telefon- tastatur).
2. Sobald diese Zahl von der Programmiersoftware erkannt wird, werden Sie aufgefordert, den Befehl zum Löschen des Speichers der Steuerung zu bestätigen.
3. Wenn Sie auf diese Frage mit „yes“ (ja) antworten, weist die Programmiersoftware die Steuerung an, den Programmspeicher zu löschen.

## Zukünftigen Zugriff zulassen (OEM-Sperre)

Die Steuerung unterstützt eine Funktion, mit der Sie festlegen können, ob der zukünftige Zugriff auf das Anwenderprogramm nach der Über- tragung auf die Steuerung zugelassen wird. Dieser Schutztyp ist vor allem für OEMs (Original Equipment Manufacturer) nützlich, die eine Anwendung entwickeln und diese anschließend über ein Speicher- modul oder innerhalb einer Steuerung verteilen.

Die Funktion „Allow Future Access“ (Zukünftigen Zugriff zulassen) finden Sie in dem Fenster „Controller Properties“ (Eigenschaften der Steuerung).




Ist die Funktion „Allow Future Access“ (Zukünftigen Zugriff zulassen) nicht ausgewählt, erfordert die Steuerung, dass das Anwender- programm in der Steuerung mit dem Programm im Programmiergerät identisch ist. Wenn dies nicht der Fall ist, wird der Zugriff auf das Anwenderprogramm in der Steuerung verweigert. Für den Zugriff auf das Anwenderprogramm löschen Sie den Speicher der Steuerung und laden das Programm erneut.

**HINWEIS**

Funktionen wie „Change Mode“ (Modusänderung), „Clear Memory“ (Speicher löschen), „Restore Program“ (Programm wiederherstellen) und „Transfer Memory Module“ (Übertragungen auf Speichermodul)“ stehen hiervon unabhängig weiterhin zur Verfügung.

Zwischen den Kennwörtern für die Steuerung und der Funktion „Allow Future Access“ (Zukünftigen Zugriff zulassen) besteht kein Zusammenhang.

# Funktionsfiles

In diesem Kapitel werden die Funktionsfiles der Steuerung beschrieben. Dabei werden folgende Themen erläutert:


- „Überblick“ auf Seite 3-2
- „Echtzeituhr-Funktionsfile“ auf Seite 3-3
- „Funktionsfile mit Einstellpotentiometerdaten“ auf Seite 3-6
- „Funktionsfile mit Speichermoduldaten“ auf Seite 3-7
- „DAT-Funktionsfile (nur MicroLogix 1500)“ auf Seite 3-10
- „Basis-Hardware-Information-Funktionsfile“ auf Seite 3-13
- „Kommunikations-Status-File“ auf Seite 3-14
- „Ein-/Ausgangsstatusfile“ auf Seite 3-19

# Überblick

Funktionsfiles bilden zusammen mit den Programmfiles und den Datenfiles die drei wichtigsten Filestrukturen bei MicroLogix 1200- und MicroLogix 1500-Steuerungen. Über Funktionsfiles wird eine effiziente und logische Schnittstelle zu Steuerungsressourcen bereitgestellt. Zu diesen Steuerungsressourcen zählen resident geladene (permanente) Funktionen wie die Echtzeituhr und der Hochgeschwindigkeitszähler. Diese Funktionen stehen dem Steuerungsprogramm entweder über Befehle für spezifische Funktionsfiles oder über Standardbefehle wie MOV und ADD zur Verfügung. Folgende Funktionsfiles werden verwendet:

**Tabelle 3.1 Funktionsfiles**

| Filename | Filekennung | Filebeschreibung |
|---|---|---|
| Hochgeschwindigkeits-zähler | **HSC** | Dieser Filtetyp wird im Zusammenhang mit dem Hochgeschwindigkeitszähler verwendet. Weitere Informationen hierzu finden Sie unter „Verwenden des Hochgeschwindigkeitszählers und des programmierbaren Endschalters“ auf Seite 5-1. |
| Frequenzausgang (Nur MicroLogix 1200- und 1500 BXB-Geräte.) | **PTO** | Dieser Filetyp wird im Zusammenhang mit dem Befehl zum Frequenzausgang verwendet. Weitere Informationen hierzu finden Sie unter „PTO-Funktionsfile (Impulsausgang)“ auf Seite 6-6. |
| Pulsweitenmodulation (Nur MicroLogix 1200- und 1500 BXB-Geräte.) | **PWM** | Dieser Filetyp wird im Zusammenhang mit dem Befehl zur Pulsweitenmodulation verwendet. Weitere Informationen hierzu finden Sie unter „Funktionsfile für Pulsweitenmodulation (PWM)“ auf Seite 6-20. |
| Wählbarer zeitgesteuerter Interrupt | **STI** | Dieser Filtetyp wird im Zusammenhang mit dem Befehl für wählbare zeitgesteuerte Interrupts verwendet. Weitere Informationen hierzu finden Sie unter „STI-Funktionsfile verwenden“ auf Seite 18-12. |
| Ereigniseingangs-Interrupt | **EII** | Dieser Filetyp wird im Zusammenhang mit dem Befehl zum Ereigniseingangs-Interrupt verwendet. Weitere Informationen hierzu finden Sie unter „EII-Funktionsfile verwenden“ auf Seite 18-17. |
| Echtzeituhr | **RTC** | Dieser Filtetyp wird im Zusammenhang mit dem Befehl für die Echtzeituhr verwendet. Weitere Informationen hierzu finden Sie unter „Echtzeituhr-Funktionsfile“ auf Seite 3-3. |
| Einstellpotentiometerdaten | **TPI** | Dieser Filetyp enthält Informationen zu Einstellpotentiometern. Weitere Informationen hierzu finden Sie unter „Funktionsfile mit Einstellpotentiometerdaten“ auf Seite 3-6. |
| Speichermoduldaten | **MMI** | Dieser Filetyp enthält Informationen zum Speichermodul. Weitere Informationen hierzu finden Sie unter „Funktionsfile mit Speichermoduldaten“ auf Seite 3-7. |
| Datenzugriff-Terminal-Information (nur MicroLogix 1500.) | **DAT** | Dieser Filetyp enthält Informationen zum Datenzugriff-Terminal. Weitere Informationen hierzu finden Sie unter „DAT-Funktionsfile (nur MicroLogix 1500)“ auf Seite 3-10. |
| Basis-Hardware-Information | **BHI** | Dieser Filetyp enthält Informationen zur Hardware der Steuerung. Informationen zur Struktur dieses Files finden Sie unter „Basis-Hardware-Information-Funktionsfile“ auf Seite 3-13. |
| Kommunikations-Status-File | **CS** | Dieser Filetyp enthält Informationen zur Kommunikation mit der Steuerung. Informationen zur Struktur dieses Files finden Sie unter „Kommunikations-Status-File“ auf Seite 3-14. |
| E/A-Status-File | **IOS** | Dieser Filetyp enthält Informationen zur Steuerungs-E/A. Informationen zur Struktur dieses Files finden Sie unter „Ein-/Ausgangsstatusfile“ auf Seite 3-19. |

# Echtzeituhr-Funktionsfile

Die Echtzeituhr liefert Datums- und Zeitangaben (Jahr, Monat, Tag, Wochentag, Stunden, Minuten, Sekunden) an den Funktionsfile RTC (Echtzeituhr) in der Steuerung.

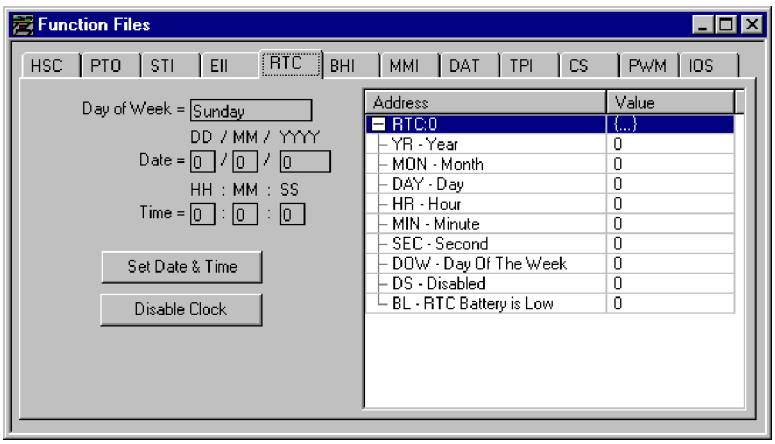
Die Echtzeituhr-Parameter und deren gültige Bereiche sind in der nachfolgenden Tabelle dargestellt.

**Tabelle 3.2 Funktionsfile Echtzeituhr**

| Angabe | Adresse | Datenformat | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| YR - RTC Jahr | RTC:0.YR | Wort | 1998 bis 2097 | Status | Nur Lesen |
| MON - RTC Monat | RTC:0.MON | Wort | 1 bis 12 | Status | Nur Lesen |
| DAY - RTC Tag | RTC:0.DAY | Wort | 1 bis 31 | Status | Nur Lesen |
| HR - RTC Stunden | RTC:0.HR | Wort | 0 bis 23 (24-Stunden-Format) | Status | Nur Lesen |
| MIN - RTC Minuten | RTC:0.MIN | Wort | 0 bis 59 | Status | Nur Lesen |
| SEC - RTC Sekunden | RTC:0.SEC | Wort | 0 bis 59 | Status | Nur Lesen |
| DOW - RTC Wochentag | RTC:0.DOW | Wort | 0 bis 6 (Sonntag bis Samstag) | Status | Nur Lesen |
| DS - deaktiviert | RTC:0/DS | Binärwert | 0 oder 1 | Status | Nur Lesen |
| BL - RTC Batterieladezustand niedrig | RTC:0/BL | Binärwert | 0 oder 1 | Status | Nur Lesen |

## Daten an die Echtzeituhr übertragen

Das Programmierfenster ist nachfolgend abgebildet:




Beim Senden gültiger Daten vom Programmiergerät oder einer anderen Steuerung an die Echtzeituhr werden die neuen Werte sofort wirksam. Klicken Sie in RSLogix 500 im Bildschirm für den RTC- Funktionsfile auf die Option *Set Date and Time* (Zeit und Datum einstellen), um die Zeit der Echtzeituhr auf die aktuelle Zeit Ihres Computers zu setzen.

Die Echtzeituhr lässt das Laden oder Speichern ungültiger Datums- oder Zeitinformationen nicht zu.

**HINWEIS** Verwenden Sie die Schaltfläche *Disable clock* (Uhr deaktivieren) Ihres Programmiergeräts, um die Echt- zeituhr zu deaktivieren, bevor Sie ein Modul ein- lagern. Damit wird die Senkung des Batterielade- zustands während der Lagerphase minimiert.

## Genauigkeit der Echtzeituhr

Die nachfolgende Tabelle zeigt die berechnete Genauigkeit der Echtzeituhr bei verschiedenen Temperaturen:

**Tabelle 3.3 Genauigkeit der Echtzeituhr bei verschiedenen Temperaturen**

| Umgebungstemperatur | Genauigkeit(1) |
|---|---|
| 0 °C | +34 bis -70 Sekunden/Monat |
| +25 °C | +36 bis -68 Sekunden/Monat |
| +40 °C | +29 bis -75 Sekunden/Monat |
| +55 °C | -133 bis -237 Sekunden/Monat |

(1) Diese Zahlen geben die ungünstigste Genauigkeit während eines Monats mit 31 Tagen an.

## RTC-Batteriebetrieb

Die Echtzeituhr ist mit einer integrierten Batterie ausgestattet, die nicht austauschbar ist. Der Funktionsfile der Echtzeituhr enthält ein Anzeige-Bit für den Batterieladezustand (RTC:0/BL) der Echtzeituhr. Bei niedrigem Batterieladezustand wird dieses Anzeige-Bit gesetzt (1). Dies bedeutet, dass die Kapazität der Batterie nach maximal 14 Tagen erschöpft sein wird und die Echtzeituhr ersetzt werden muss. Wenn das Anzeige-Bit für den Ladezustand nicht gesetzt ist (0), ist der Ladezustand der Batterie in Ordnung oder es wurde keine Echtzeituhr angeschlossen.

**ACHTUNG**

Wenn die Echtzeituhr bei niedrigem Batterielade- zustand länger als 14 Tage betrieben wird, ist die Korrektheit der Echtzeituhrdaten nicht mehr gewährleistet, wenn die Spannungsversorgung der Steuerung unterbrochen wird.

**Tabelle 3.4 Lebensdauer der Batterie der Echzeituhr**

| Status der Batterie | Temperatur | Dauer |
|---|---|---|
| Betrieb | 0 °C bis +40 °C | 5 Jahre(1) |
| Lagerung | -40 °C bis +25 °C | mind. 5 Jahre |
| | +26 °C bis +60 °C | mind. 3 Jahre |

(1) Die Betriebsdauer der Batterie basiert auf einer vorhergehenden Lagerung von sechs Monaten.

# RTA-Befehl (Echtzeituhr anpassen)

Befehlstyp: Ausgang

**Ausführungszeit des RTA-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 4,7 µs | 3,7 µs |
| | 556,2 µs (wahr-zu-unwahr-Wechsel) | |
| MicroLogix 1500 | 4,1 µs | 2,6 µs |
| | 426,8 µs (wahr-zu-unwahr-Wechsel) | |

Der RTA-Befehl wird zum Synchronisieren der Echtzeituhr (RTC) der Steuerung mit einer externen Quelle verwendet. Mithilfe des RTA-Befehls wird die RTC bis auf die Minute genau angepasst. Mithilfe des RTA-Befehls wird die RTC basierend auf dem Wert der RTC-Sekunden angepasst, wie unten näher beschrieben.

**WICHTIG** Die RTC wird nur mithilfe des RTA-Befehls geändert, wenn der RTA-Strompfad als wahr erkannt wird, nachdem er vorher unwahr gewesen ist (unwahr-zu- wahr-Wechsel). Der RTA-Befehl hat keine Wirkung, wenn der Strompfad immer wahr oder immer unwahr ist.

RTA wurde gesetzt:

- Wenn der Wert der RTC-Sekunden kleiner als 30 ist, werden die RTC-Sekunden auf 0 zurückgesetzt.
- Wenn der Wert der RTC-Sekunden größer als oder gleich 30 ist, werden der Wert für die RTC-Minuten um 1 erhöht und die RTC-Sekunden auf 0 zurückgesetzt.

Unter den folgenden Bedingungen hat der RTA-Befehl keine Auswirkungen auf die Daten der RTC:

- Es ist keine RTC an die Steuerung angeschlossen
- Eine RTC ist angeschlossen, jedoch deaktiviert
- Es wird eine externe Meldung an die RTC (über Kommunikations- systeme) bearbeitet, während der RTA-Befehl ausgeführt wird. (Externe Meldungen an die RTC haben Vorrang vor RTA- Befehlen.)

Um den RTA-Befehl wieder zu aktivieren, muss der RTA-Strompfad zuerst unwahr und anschließend wahr werden.

**HINWEIS** Im System wird nur ein internes Speicher-Bit für diesen Befehl zugeordnet. Verwenden Sie in Ihrem Programm nicht mehr als einen RTA-Befehl.

**HINWEIS** Verwenden Sie einen MSG-Befehl, um die RTC-Daten von einer Steuerung auf eine andere zu schreiben und die Zeit zu synchronisieren. Zum Senden (Schreiben) von RTC-Daten verwenden Sie als Quelle RTC:0. *Diese Funktion steht für Steuerungen der Serie A nicht zur Verfügung.*

## Funktionsfile mit Einstellpotentiometerdaten

Der Funktionsfile mit den Einstellpotentiometerdaten (TPI) enthält folgende Angaben:

**Tabelle 3.5 Funktionsfile mit Einstellpotentiometer**

| Daten | Adresse | Datenformat | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| TPD-Daten 0 | TPI:0.POT0 | Wort (ganzzahlige 16-Bit-Daten mit Vorzeichen) | 0 - 250 | Status | Nur Lesen |
| TPD-Daten 1 | TPI:0.POT1 | Wort (ganzzahlige 16-Bit-Daten mit Vorzeichen) | 0 - 250 | Status | Nur Lesen |
| TP0 Fehlercode | TPI:0.ER | Wort (Bits 0 bis 7) | 0 - 3 | Status | Nur Lesen |
| TP1 Fehlercode | | Wort (Bits 8 bis 15) | | | |

Die in TPI:0.POT0 enthaltenen Daten stellen die Position des Einstellpotentiometers 0 dar. Die in TPI:0.POT1 enthaltenen Daten entsprechen der Position des Einstellpotentiometers 1. Der gültige Datenbereich für beide Werte liegt zwischen 0 (gegen den Uhr-zeigersinn) bis 250 (im Uhrzeigersinn).

## Fehlerzustände

Wenn die Steuerung ein Problem bei einem der Einstellpotentiometer feststellt, bleiben die zuletzt gelesenen Werte an der Datenadresse; außerdem wird in dem Fehlercodebyte der TPI-Datei für das Einstellpotentiometer, an dem das Problem auftrat, ein Fehlercode generiert. Sobald die Steuerung Zugriff auf die Hardware des Einstellpotentiometers hat, wird der Fehlercode gelöscht. Die Fehlercodes werden in der nachfolgenden Tabelle beschrieben.

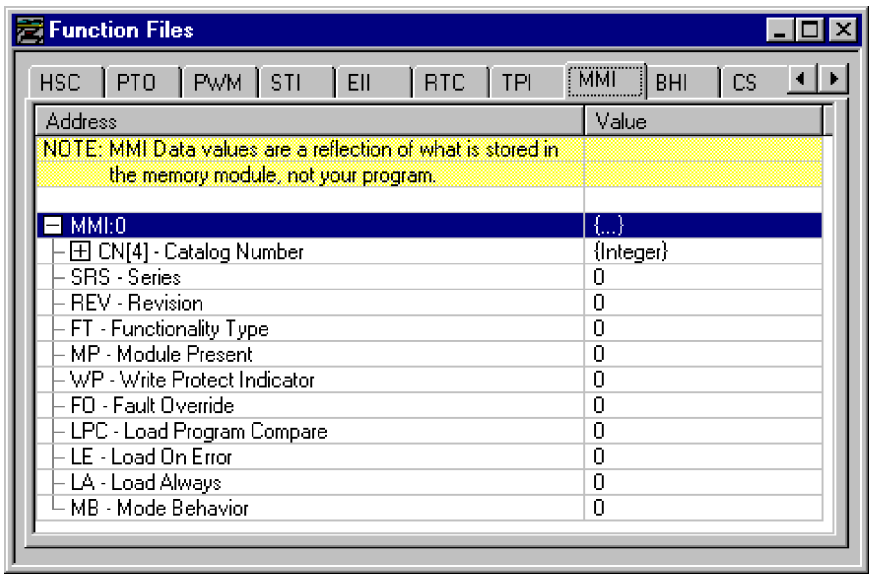
**Tabelle 3.6 Fehlercodes für Einstellpotentiometer**

| Fehlercode | Beschreibung |
|---|---|
| 0 | Daten des Einstellpotentiometers gültig. |
| 1 | Subsystem des Einstellpotentiometers erkannt, Daten ungültig. |
| 2 | Subsystem des Einstellpotentiometers konnte nicht initialisiert werden. |
| 3 | Fehler in Subsystem des Einstellpotentiometers. |

## Funktionsfile mit Speichermoduldaten

Die Steuerung enthält einen MMI-File (File mit Speichermoduldaten), das durch Daten aus dem angeschlossenen Speichermodul aktualisiert wird. Beim Einschalten oder bei Einbau eines Speichermoduls werden die Bestellnummer, die Serie, die Revision und der Typ (Speicher- modul und/oder Echtzeituhr) erkannt und in den MMI-File im Anwen- derprogramm geschrieben. Wenn das Speichermodul und/oder die Echtzeituhr nicht angeschlossen ist, werden Nullwerte in den MMI-File geschrieben.

Das Programmierfenster für den MMI-Funktionsfile ist nachfolgend abgebildet:




Die Parameter und deren gültige Bereiche sind in der nachfolgenden Tabelle dargestellt.

**Tabelle 3.7 Parameter des MMI-Funktionsfiles**

| Angabe | Adresse | Datenformat | Typ | Anwender-programmzugriff |
|---|---|---|---|---|
| FT - Funktionstyp | MMI:0.FT | Wort (INT) | Status | Nur Lesen |
| MP - Modul vorhanden | MMI:0/MP | Binärwert (Bit) | Status | Nur Lesen |
| WP - Schreibschutz | MMI:0/WP | Binärwert (Bit) | Steuerung | Nur Lesen |
| FO - Fehler überbrücken | MMI:0/FO | Binärwert (Bit) | Steuerung | Nur Lesen |
| LPC - Programmvergleich | MMI:0/LPC | Binärwert (Bit) | Steuerung | Nur Lesen |
| LE - bei Fehler laden | MMI:0/LE | Binärwert (Bit) | Steuerung | Nur Lesen |
| LA - immer laden | MMI:0/LA | Binärwert (Bit) | Steuerung | Nur Lesen |
| MB - Modusverhalten | MMI:0/MB | Binärwert (Bit) | Steuerung | Nur Lesen |

### *FT – Funktionstyp*

Das LSB dieses Worts bezeichnet den installierten Modultyp:

- 1 = Speichermodul
- 2 = Echtzeituhr
- 3 = Speichermodul und Echtzeituhr

### *MP – Modul vorhanden*

Das MP-Bit (Modul vorhanden) kann in dem Anwenderprogramm ver- wendet werden, um festzustellen, ob ein Speichermodul auf der Speicherung vorhanden ist. Dieses Bit wird einmal pro Abfrage- vorgang aktualisiert, sofern das Speichermodul grundsätzlich von der Steuerung erkannt wurde. Diese erstmalige Erkennung durch die Steuerung erfolgt nur, wenn das Speichermodul vor dem Einschalten der Steuerung oder während eines nicht ausführenden Modus der Steuerung installiert wurde. Wird ein Speichermodul installiert, während sich die Steuerung in einem Ausführungsmodus befindet, so wird es nicht erkannt. Wenn ein bereits erkanntes Speichermodul während eines Ausführungsmodus ausgebaut wird, wird dieses Bit am Ende der nächsten Kontaktplanabfrage gelöscht (0).

### *WP – Schreibschutz*

Wenn das WP-Bit (Schreibschutz) gesetzt wurde (1), ist das Modul schreibgeschützt, und das Anwenderprogramm und die Daten innerhalb des Moduls können nicht überschrieben werden.

**WICHTIG** Das WP-Bit kann nach dem Setzen (1) nicht mehr gelöscht werden. Setzen Sie dieses Bit nur, wenn der Inhalt des Speichermoduls dauerhaft geschützt werden soll.

### *FO – Fehler überbrücken*

Das FO-Bit (Fehler überbrücken) zeigt den Status der Auswahl- möglichkeit „Fault override“ (Fehler überbrücken) in dem Programm an, das in dem Speichermodul gespeichert ist. Auf diese Weise kann der Wert des FO-Bits geprüft werden, ohne dass das Anwender- programm aus dem Speichermodul geladen werden muss.

**WICHTIG** Das Betriebsverhalten der Steuerung wird nicht durch die Einstellung „Override selection“ (Fehler überbrücken) des Speichermoduls in dem MMI-File festgelegt. Dieses Bit zeigt nur die Einstellung des Fehler-überbrücken-Bits (S:1/8) in dem Anwender- programm in dem Speichermodul an.

Weitere Informationen hierzu finden Sie unter „Fehlerüberbrückung beim Einschalten“ auf Seite C-5.

### *LPC – Programmvergleich*

Das LPC-Bit (Programmvergleich) zeigt den Status der Auswahl- möglichkeit „Load Program Compare“ (Programmvergleich) in dem Anwenderprogramm-Statusfile des Speichermoduls an. Auf diese Weise lässt sich der Wert bestimmen, ohne das Anwenderprogramm tatsächlich vom Speichermodul laden zu müssen.

Weitere Informationen hierzu finden Sie unter „Speichermodul-Programmvergleich“ auf Seite C-10.

### *LE – Bei Fehler laden*

Das LE-Bit (Bei Fehler laden) zeigt den Status der Auswahlmöglichkeit „Load on Error“ (Bei Fehler laden) in dem Programm an, das in dem Speichermodul

gespeichert ist. Auf diese Weise kann der Wert der Auswahl geprüft werden, ohne dass das Anwenderprogramm aus dem Speichermodul geladen werden muss.

Weitere Informationen hierzu finden Sie unter „Speichermodul bei Fehler oder Standardprogramm laden“ auf Seite C-6.

### *LA – Immer laden*

Das LA-Bit (Immer laden) zeigt den Status der Auswahlmöglichkeit „Load Always“ (Immer laden) in dem Programm an, das in dem Speichermodul gespeichert ist. Auf diese Weise kann der Wert der Auswahl geprüft werden, ohne dass das Anwenderprogramm aus dem Speichermodul geladen werden muss.

Weitere Informationen hierzu finden Sie unter „Speichermodul immer laden“ auf Seite C-6.

### *MB – Modusverhalten*

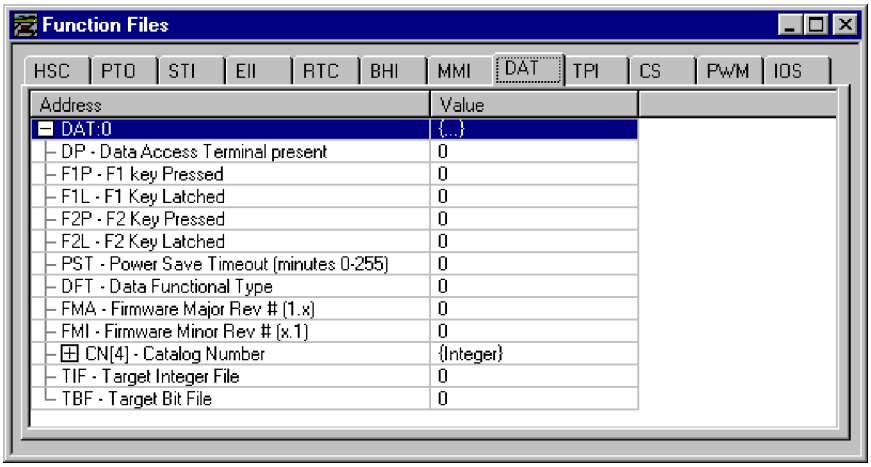
Das MB-Bit (Modusverhalten) zeigt den Status der Auswahlmöglich- keit „Mode Behavior“ (Modusverhalten) in dem Programm an, das in dem Speichermodul gespeichert ist. Auf diese Weise kann der Wert der Auswahl geprüft werden, ohne dass das Anwenderprogramm aus dem Speichermodul geladen werden muss.

Weitere Informationen hierzu finden Sie unter „Einschaltmodusverhalten“ auf Seite C-7.

# DAT-Funktionsfile (nur MicroLogix 1500)

**HINWEIS** Dieser Abschnitt beschreibt den DAT-Funktionsfile. Angaben zur Arbeit mit DAT finden Sie im Benutzerhandbuch *Speicherprogrammierbare Steuerungen MicroLogix™ 1500*, Publikation 1764-UM001).

Die DAT-Konfiguration (Datenzugriffsmodul) ist in dem Prozessor in einer separaten Datei, dem DAT-Funktionsfile, gespeichert. Der DAT- Funktionsfile ist Teil des Steuerprogramms des Anwenders (siehe unten).




Der Funktionsfile DAT enthält den Ziel-Ganzzahl-File, den Ziel-Bit-File und den PST-Parameter (Energiesparfunktion). Diese drei Parameter werden in der nachfolgenden Tabelle beschrieben.

| Angabe | Adresse | Datenformat | Typ | Anwender-programmzugriff |
|---|---|---|---|---|
| Ziel-Ganzzahl-File | DAT:0.TIF | Wort (INT) | Steuerung | Nur Lesen |
| Ziel-Bit-File | DAT:0.TBF | Wort (INT) | Steuerung | Nur Lesen |
| Energiesparfunktion (PST, Power Save Timeout) | DAT:0.PST | Wort (INT) | Steuerung | Nur Lesen |

## Ziel-Ganzzahl-File (TIF)

Die Werte, die an dem TIF-Standort gespeichert werden, kennzeichnen den Ganzzahl-File, zu dem das DAT eine Verbindung aufbaut. Das DAT kann jeden gültigen Ganzzahl-File innerhalb der Steuerung lesen bzw. in diesen File schreiben. Gültige Ganzzahl-Files sind N3 bis N255. Wenn das DAT eine gültige Ganzzahl-Filenummer liest, kann es die ersten 48 Elemente (0 bis 47) dieses Files in der DAT-Anzeige darstellen. Über die nächsten 48 Bits (Worte 48 bis 50) werden die Zugriffsberechtigungen (Nur-Lesen oder Lesen/Schreiben) für die 48 Elemente definiert.

Der einzige Ganzzahl-File, der eine DAT-Schnittstelle bietet, ist der am TIF-Standort angegebene File. Der TIF-Standort kann nur durch Herunterladen eines Programms geändert werden.

**WICHTIG** Stellen Sie mit Hilfe Ihrer Programmiersoftware sicher, dass der Ganzzahl-File, den Sie an dem TIF-Standort angeben, sowie die entsprechende Anzahl an Elementen in dem Anwenderprogramm der Steuerung vorhanden sind.

Die nachfolgende Beispieltabelle zeigt die Konfiguration eines DAT, das die Ganzzahl-Filenummer 50 (DAT:0.TIF = 50) verwendet.

| Element-nummer | Datenadresse | Schutz-Bit | Element-nummer | Datenadresse | Schutz-Bit | Element-nummer | Datenadresse | Schutz-Bit |
|---|---|---|---|---|---|---|---|---|
| 0 | N50:0 | N50:48/0 | 16 | N50:16 | N50:49/0 | 32 | N50:32 | N50:50/0 |
| 1 | N50:1 | N50:48/1 | 17 | N50:17 | N50:49/1 | 33 | N50:33 | N50:50/1 |
| 2 | N50:2 | N50:48/2 | 18 | N50:18 | N50:49/2 | 34 | N50:34 | N50:50/2 |
| 3 | N50:3 | N50:48/3 | 19 | N50:19 | N50:49/3 | 35 | N50:35 | N50:50/3 |
| 4 | N50:4 | N50:48/4 | 20 | N50:20 | N50:49/4 | 36 | N50:36 | N50:50/4 |
| 5 | N50:5 | N50:48/5 | 21 | N50:21 | N50:49/5 | 37 | N50:37 | N50:50/5 |
| 6 | N50:6 | N50:48/6 | 22 | N50:22 | N50:49/6 | 38 | N50:38 | N50:50/6 |
| 7 | N50:7 | N50:48/7 | 23 | N50:23 | N50:49/7 | 39 | N50:39 | N50:50/7 |
| 8 | N50:8 | N50:48/8 | 24 | N50:24 | N50:49/8 | 40 | N50:40 | N50:50/8 |
| 9 | N50:9 | N50:48/9 | 25 | N50:25 | N50:49/9 | 41 | N50:41 | N50:50/9 |
| 10 | N50:10 | N50:48/10 | 26 | N50:26 | N50:49/10 | 42 | N50:42 | N50:50/10 |
| 11 | N50:11 | N50:48/11 | 27 | N50:27 | N50:49/11 | 43 | N50:43 | N50:50/11 |
| 12 | N50:12 | N50:48/12 | 28 | N50:28 | N50:49/12 | 44 | N50:44 | N50:50/12 |
| 13 | N50:13 | N50:48/13 | 29 | N50:29 | N50:49/13 | 45 | N50:45 | N50:50/13 |
| 14 | N50:14 | N50:48/14 | 30 | N50:30 | N50:49/14 | 46 | N50:46 | N50:50/14 |
| 15 | N50:15 | N50:48/15 | 31 | N50:31 | N50:49/15 | 47 | N50:47 | N50:50/15 |

Die Elementnummer, die auf dem DAT angezeigt wird, entspricht dem in der Tabelle dargestellten Datenregister. Das Schutz-Bit legt fest, ob die Daten schreibgeschützt sind oder ein Lese-/Schreibzugriff besteht. Wenn das Schutz-Bit gesetzt ist (1), hat das DAT nur einen Lesezugriff auf die entsprechende Datenadresse. Die LED „Protected“ (Geschützt) leuchtet auf, wenn ein schreibgeschütztes Element auf der DAT- Anzeige aktiv ist. Wenn das Schutz-Bit gelöscht (0) oder nicht vorhanden ist, leuchtet die LED „Protected“ nicht auf, und die Daten der entsprechenden Adresse können über die DAT-Tastatur bearbeitet werden.

**WICHTIG** Obwohl geschützte Daten nicht über die Tastatur des DAT geändert werden können, haben das Steuerprogramm oder andere Kommunikations- geräte Zugriff auf diese Daten. Schutz-Bits stellen keinen Überschreibschutz für Daten innerhalb des Ziel-Ganzzahl-Files dar. Es liegt vollständig in der Verantwortung des Anwenders, sicherzustellen, dass Daten nicht unbeabsichtigt überschrieben werden.

**HINWEIS**

- Die übrigen Adressen in dem Ziel-File können ohne Einschränkung verwendet werden (in diesem Beispiel die Adressen N50:51 und höher).
- Das DAT beginnt immer mit Wort 0 eines Datenfiles. Ein Beginn an einer anderen Adresse innerhalb des Files ist nicht möglich.

## Ziel-Bit-File (TBF)

Die Werte, die an dem TBF-Standort gespeichert werden, kenn-zeichnen den Bit-File, zu dem das DAT eine Verbindung aufbaut. Das DAT kann jeden gültigen Bit-File innerhalb der Steuerung lesen bzw. in diesen File schreiben. Gültige Bit-Files sind B3 bis B255. Wenn das DAT eine gültige Bit-File-Nummer liest, kann es im Anzeigebildschirm auf die ersten 48 Bits (0 bis 47) des angegebenen Files zugreifen. Über die nächsten 48 Bits (48 bis 95) werden die Zugriffsberechtigungen (Nur-Lesen oder Lesen/Schreiben) für die ersten 48 Bits definiert.

Das DAT stellt nur zu dem Bit-File, das an dem TBF-Standort genannt wird, eine Verbindung her. Der TBF-Standort kann nur durch Herunterladen eines Programms geändert werden.

**WICHTIG** Stellen Sie mit Hilfe Ihrer Programmiersoftware sicher, dass der Bit-File, den Sie an dem TBF-Standort angeben, sowie die entsprechende Anzahl an Elementen in dem MicroLogix 1500-Anwender- programm vorhanden sind.

Die nachfolgende Beispieltabelle zeigt die Verwendung der Konfigurationsangaben für die Bit-File-Nummer 51 (DAT:0.TBF=51) durch das DAT.

| Bitnummer | Datenadresse | Schutz-Bit | Bitnummer | Datenadresse | Schutz-Bit | Bitnummer | Datenadresse | Schutz-Bit |
|---|---|---|---|---|---|---|---|---|
| 0 | B51/0 | B51/48 | 16 | B51/16 | B51/64 | 32 | B51/32 | B51/80 |
| 1 | B51/1 | B51/49 | 17 | B51/17 | B51/65 | 33 | B51/33 | B51/81 |
| 2 | B51/2 | B51/50 | 18 | B51/18 | B51/66 | 34 | B51/34 | B51/82 |
| 3 | B51/3 | B51/51 | 19 | B51/19 | B51/67 | 35 | B51/35 | B51/83 |
| 4 | B51/4 | B51/52 | 20 | B51/20 | B51/68 | 36 | B51/36 | B51/84 |
| 5 | B51/5 | B51/53 | 21 | B51/21 | B51/69 | 37 | B51/37 | B51/85 |
| 6 | B51/6 | B51/54 | 22 | B51/22 | B51/70 | 38 | B51/38 | B51/86 |
| 7 | B51/7 | B51/55 | 23 | B51/23 | B51/71 | 39 | B51/39 | B51/87 |
| 8 | B51/8 | B51/56 | 24 | B51/24 | B51/72 | 40 | B51/40 | B51/88 |
| 9 | B51/9 | B51/57 | 25 | B51/25 | B51/73 | 41 | B51/41 | B51/89 |
| 10 | B51/10 | B51/58 | 26 | B51/26 | B51/74 | 42 | B51/42 | B51/90 |
| 11 | B51/11 | B51/59 | 27 | B51/27 | B51/75 | 43 | B51/43 | B51/91 |
| 12 | B51/12 | B51/60 | 28 | B51/28 | B51/76 | 44 | B51/44 | B51/92 |
| 13 | B51/13 | B51/61 | 29 | B51/29 | B51/77 | 45 | B51/45 | B51/93 |
| 14 | B51/14 | B51/62 | 30 | B51/30 | B51/78 | 46 | B51/46 | B51/94 |
| 15 | B51/15 | B51/63 | 31 | B51/31 | B51/79 | 47 | B51/47 | B51/95 |

Die Bitnummer, die auf dem DAT angezeigt wird, entspricht dem in der Tabelle dargestellten Daten-Bit. Das Schutz-Bit legt fest, ob die Daten schreibgeschützt sind oder ein Lese-/Schreibzugriff besteht. Wenn das Schutz-Bit gesetzt ist (1), hat das DAT nur einen Lesezugriff auf die entsprechende Datenadresse. Die LED „Protected“ (Geschützt) leuchtet auf, wenn ein schreibgeschütztes Element auf der DAT- Anzeige aktiv ist. Wenn das Schutz-Bit gelöscht (0) oder nicht vorhanden ist, leuchtet die LED „Protected“ nicht auf, und die Daten der entsprechenden Adresse können über die DAT-Tastatur bearbeitet werden.

**WICHTIG** Obwohl geschützte Daten nicht über die Tastatur des DAT geändert werden können, haben das Steuer- programm oder andere Kommunikationsgeräte Zugriff auf diese Daten. Schutz-Bits stellen keinen Überschreibschutz für Daten innerhalb des Ziel-Bit- Files dar. Es liegt vollständig in der Verantwortung des Anwenders, sicherzustellen, dass Daten nicht unbeabsichtigt überschrieben werden.

**HINWEIS**

- Die übrigen Adressen in dem Ziel-File können ohne Einschränkung verwendet werden (in diesem Beispiel die Adressen B51/96 und höher).
- Das DAT beginnt immer mit Bit 0 eines Datenfiles. Ein Beginn an einer anderen Adresse innerhalb des Files ist nicht möglich.

## Basis-Hardware-Information-Funktionsfile

Der BHI-File (Basis-Hardware-Information) ist ein Nur-Lesen-File, der eine Beschreibung der MicroLogix 1200-Steuerung oder des MicroLogix 1500-Basisgerät enthält.

**Tabelle 3.8 Funktionsfile Basis-Hardware-Information (BHI)**

| Adresse | Beschreibung |
|---|---|
| BHI:0.CN | CN - Bestellnummer |
| BHI:0.SRS | SRS - Serie |
| BHI:0.REV | REV - Revision |
| BHI:0.FT | FT - Funktionstyp |

## Kommunikations-Status-File

Der Kommunikations-Status-File (CS) ist ein Nur-Lesen-File, der Informationen zur Konfiguration der Parameter für die Steuerungskommunikation sowie Statusinformationen zu Kommunikations- vorgängen enthält.

Der Kommunikations-Status-File verwendet:

**Tabelle 3.9 Größe des Kommunikations-Status-Files**

| Steuerung | Anzahl der Wortelemente |
|---|---|
| MicroLogix 1500 1764-LSP-Prozessoren der Serie A | 44 1-Wort-Elemente |
| MicroLogix 1200<br>MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren | 71 1-Wort-Elemente |

Es gibt für jeden Kommunikationsanschluss einen Kommunikations- Status-File. Kommunikations-Status-File CS0 entspricht Kanal 0 der Steuerung. Kommunikations-Status-File CS1 entspricht Kanal 1 des 1764-LRP-Prozessors.

**HINWEIS** Sie können den Kommunikations-Status-File zur Suche von Kommunikationsfehlern einsetzen.

Der Datenfile hat folgende Struktur:

**Tabelle 3.10 Kommunikations-Status-File**

| Wort | Beschreibung | Gilt für Steuerung | Einzelheiten auf Seite |
|---|---|---|---|
| 0 bis 5 | Block zum allgemeinen Kanalstatus | MicroLogix 1200 und 1500 | 3-15 |
| 6 bis 22 | DLL-Diagnosezählerblock | MicroLogix 1200 und 1500 | 3-15 |
| 23 bis 42 | Block zur aktiven DLL-Netzknotentabelle | MicroLogix 1200 und 1500 | 3-18 |
| *Die Worte 43 bis 70, wenn DF1-Vollduplex, DF1-Halbduplex, DH-485 oder ASCII verwendet wird (1):* | | | |
| 43 | Kennung für Listenende (immer 0) | MicroLogix 1200 und 1500 | -- |
| 43 bis 70 | Reserviert | • MicroLogix 1200<br>• MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren | -- |
| *Wörter 43 bis 70 bei Verwendung von Modbus RTU Slave:* | | | |
| 43 bis 69 | Modbus Slave Diagnosezählerblock | • MicroLogix 1200<br>• MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren | 3-18 |
| 70 | Kennung für Listenende (immer 0) | • MicroLogix 1200<br>• MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren | -- |

(1) ASCII kann nur mit MicroLogix 1200 und MicroLogix 1500 1764-LSP-Prozessoren der Serie B (und höher) sowie mit 1764-LRP-Prozessoren verwendet werden.

Die folgenden Tabellen enthalten Einzelheiten zu jedem Block in dem Kommunikations-Status-File.

**Tabelle 3.11 Block zum allgemeinen Kanalstatus**

| Wort | Bit | Beschreibung |
|---|---|---|
| 0 | - | Kennung für allgemeine Statusinformationen für Kommunikationskanal |
| 1 | - | Länge |
| 2 | - | Formatcode |
| 3 | - | Fehlercode für Kommunikationskonfiguration |
| 4 | 0 | ICP – Bit „Incoming Command Pending" (Eingehender Befehl anstehend)<br>Dieses Bit wird durch die Steuerung gesetzt (1), wenn ein anderes Gerät Informationen von dieser Steuerung anfordert. Sobald die angeforderten Informationen an das andere Gerät übertragen wurden, wird das Bit gelöscht (0). |
| | 1 | MRP – Bit „Incoming Message Reply Pending" (Eingehende Nachricht Antwort anstehend)<br>Dieses Bit wird durch die Steuerung gesetzt (1), sobald ein anderes Gerät die Informationen bereitgestellt hat, die durch einen MSG-Befehl angefordert wurden, der von dieser Steuerung ausgeführt wurde. Sobald der entsprechende MSG-Befehl bearbeitet wurde (bei Abfrage-Ende, SVC oder REF), wird dieses Bit gelöscht (0). |
| | 2 | MCP – Bit „Outgoing Message Command Pending" (Ausgehende Nachricht Befehl anstehend)<br>Dieses Bit wird durch die Steuerung gesetzt (1), wenn ein oder mehrere MSG-Befehle aktiviert sind und in der Kommunikationswarteschlange stehen. Dieses Bit wird gelöscht (0), sobald die Warteschlange leer ist. |
| | 3 | SSB – Bit Auswahlstatus<br>Dieses Bit zeigt an, dass sich die Steuerung im Systemmodus befindet. Dieses Bit ist immer gesetzt. |
| | 4 | CAB – Bit „Communications Active" (Aktive Kommunikation)<br>Dieses Bit wird gesetzt (1), wenn sich mindestens ein anderes Gerät im DH-485-Netzwerk befindet. Befinden sich keine anderen Geräte im Netzwerk, wird dieses Bit gelöscht (0). |
| | 5 bis 14 | Reserviert |
| | 15 | Die Umschalttaste für den Kommunikationsmodus ist standardmäßig aktiviert. Dieses Bit wird gesetzt (1), wenn sich Kanal 0 im Standardkommunikationsmodus befindet. Dieses Bit wird gelöscht (0), wenn sich Kanal 0 im anwenderkonfigurierten Kommunikationsmodus befindet. (Immer 0 für Kanal 1 des 1764-LRP-Prozessors) *Dieses Bit ist für Steuerungen der Serie A nicht verfügbar.* |
| 5 | 0 bis 7 | Netzknotenadresse - Dieses Byte enthält die Netzknotenadresse der Steuerung im Netzwerk. |
| | 8 bis 15 | Baudrate - Dieses Byte enthält die Baudrate der Steuerung im Netzwerk. |

Diagnosezählerblocks werden angezeigt bei:

- DH-485
- DF1-Vollduplex
- DF1 Halbduplex Slave
- Modbus™ RTU Slave
- ASCII

**Tabelle 3.12 DH-485-Diagnosezählerblock**

| Wort | Bit | Beschreibung |
|---|---|---|
| 6 | - | Kennung für Diagnosezähler (immer 2) |
| 7 | - | Länge (immer 30) |
| 8 | - | Formatcode (immer 0) |
| 9 | - | Summe empfangene Nachrichtenpakete |
| 10 | - | Summe übertragene Nachrichtenpakete |
| 11 | 0 bis 7 | Wiederholungen Nachrichtenpakete |
| | 8 bis 15 | max. Anzahl Wiederholungen überschritten (keine Übertragung) |

**Tabelle 3.12 DH-485-Diagnosezählerblock**

| Wort | Bit | Beschreibung |
|---|---|---|
| 12 | 0 bis 7 | NAK – keine Bestätigung übertragen |
| | 8 bis 15 | NAK – keine Bestätigung empfangen |
| 13 | 0 bis 7 | Summe ungültige empfangene Nachrichtenpakete |
| | 8 bis 15 | Reserviert |
| 14 bis 22 | - | Reserviert |

**Tabelle 3.13 DF1-Vollduplex-Diagnosezählerblock**

| Wort | Bit | Beschreibung |
|---|---|---|
| 6 | - | Kennung für Diagnosezähler (immer 2) |
| 7 | - | Länge (immer 30) |
| 8 | - | Formatcode (immer 1) |
| 9 | 0 | CTS |
| | 1 | RTS |
| | 2 | Reserviert |
| | 3 | Kanal 0 - Reserviert, Kanal 1 - DCD |
| | 4 bis 15 | Reserviert |
| 10 | - | Summe übertragene Nachrichtenpakete |
| 11 | - | Summe empfangene Nachrichtenpakete |
| 12 | - | nicht übertragene Nachrichtenpakete |
| 13 | - | übertragene ENQ-Abfragepakete |
| 14 | - | empfangene NAK-Pakete |
| 15 | - | empfangene ENQ-Abfragepakete |
| 16 | - | empfangene und nicht bestätigte ungültige Nachrichtenpakete |
| 17 | - | kein Puffer und negative Rückmeldung |
| 18 | - | empfangene doppelte Nachrichtenpakete |
| 19 bis 22 | - | Reserviert |

**Tabelle 3.14 DF1-Halbduplex-Slave-Diagnosezählerblock**

| Wort | Bit | Beschreibung |
|---|---|---|
| 6 | - | Kennung für Diagnosezähler (immer 2) |
| 7 | - | Länge (immer 30) |
| 8 | - | Formatcode (immer 2) |
| 9 | 0 | CTS |
| | 1 | RTS |
| | 2 | Reserviert |
| | 3 | Kanal 0 - Reserviert, Kanal 1 - DCD |
| | 4 bis 15 | Reserviert |
| 10 | - | Summe übertragene Nachrichtenpakete |
| 11 | - | Summe empfangene Nachrichtenpakete |
| 12 | - | nicht übertragene Nachrichtenpakete |
| 13 | - | Wiederholungen Nachrichtenpakete |
| 14 | - | empfangene NAK-Pakete |
| 15 | - | Polling empfangen |
| 16 | - | empfangene ungültige Nachrichtenpakete |

**Tabelle 3.14 DF1-Halbduplex-Slave-Diagnosezählerblock**

| Wort | Bit | Beschreibung |
|---|---|---|
| 17 | - | kein Puffer |
| 18 | - | empfangene doppelte Nachrichtenpakete |
| 19 bis 22 | - | Reserviert |

**Tabelle 3.15 Modbus RTU Slave-Diagnosezählerblock**

*(MicroLogix 1200-Steuerungen und MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren)*

| Wort | Bit | Beschreibung |
|---|---|---|
| 6 | - | Kennung für Diagnosezähler (immer 2) |
| 7 | - | Länge (immer 30) |
| 8 | - | Formatcode (immer 4) |
| 9 | 0 | CTS |
| | 1 | RTS |
| | 2 | Reserviert |
| | 3 | Kanal 0 - Reserviert, Kanal 1 - DCD |
| | 4 bis 15 | Reserviert |
| 10 | - | Summe übertragene Nachrichtenpakete |
| 11 | - | Summe empfangene Nachrichtenpakete für diesen Slave |
| 12 | - | Summe empfangene Nachrichtenpakete |
| 13 | - | Fehlerzähler Netzwerkebene |
| 14 | - | Fehlercode Netzwerkebene |
| 15 bis 22 | - | Reserviert |

**Tabelle 3.16 ASCII-Diagnosezählerblock**

*(MicroLogix 1200-Steuerungen, MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren)*

| Wort | Bit | Beschreibung |
|---|---|---|
| 6 | - | Kennung für DLL-Diagnosezähler (immer 2) |
| 7 | - | Länge (immer 30) |
| 8 | - | Formatcode (immer 5) |
| 9 | 0 | CTS |
| | 1 | RTS |
| | 2 | Reserviert |
| | 3 | Kanal 0 - Reserviert, Kanal 1 - DCD |
| | 4 bis 15 | Reserviert |
| 10 | 0 | Software-Handshaking-Status |
| | 1 bis 15 | Reserviert |
| 11 | - | Echo-Zeichenzählwert |
| 12 | - | Zählwert empfangener Zeichen |
| 13 bis 18 | - | Reserviert |
| 19 | - | Zählwert für fehlerhafte Zeichen |
| 20 bis 22 | - | Reserviert |

**Tabelle 3.17 Block aktive Netzknotentabelle**

| Wort | Beschreibung |
|---|---|
| 23 | Kennung für aktive Netzknotentabelle (immer 3) |
| 24 | Länge (immer 4 für DH-485, immer 0 für DF1-Vollduplex, DF1-Halbduplex-Slave, Modbus-RTU-Slave und ASCII) |
| 25 | Formatcode (immer 0) |
| 26 | Anzahl der Knoten (immer 32 für DH-485, immer 0 für DF1-Vollduplex, DF1-Halbduplex-Slave, Modbus-RTU-Slave und ASCII) |
| 27 | Aktive Netzknotentabelle – Netzknoten 0 bis 15 (CS0:27/1 ist Netzknoten 1, CS0:27/2 ist Netzknoten 2 usw.) Dies ist ein Register mit Bitbelegung, der den Status jedes Knotens im Netzwerk anzeigt. Wenn ein Bit gesetzt ist (1), ist der entsprechende Netzknoten im Netzwerk aktiv. Wenn ein Bit nicht gesetzt ist (0), ist der entsprechende Netzknoten nicht im Netzwerk aktiv. |
| 28 | Aktive Netzknotentabelle – Netzknoten 16 bis 31 (CS0:28/1 ist Netzknoten 16, CS0:28/2 ist Netzknoten 17 usw.) Dies ist ein Register mit Bitbelegung, der den Status jedes Knotens im Netzwerk anzeigt. Wenn ein Bit gesetzt ist (1), ist der entsprechende Netzknoten im Netzwerk aktiv. Wenn ein Bit nicht gesetzt ist (0), ist der entsprechende Netzknoten nicht im Netzwerk aktiv. |
| 29 bis 42 | Reserviert |

**Tabelle 3.18 Modbus-RTU-Slave-Diagnose**
*(MicroLogix 1200-Steuerungen, MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren)*

| Wort | Bit | Beschreibung |
|---|---|---|
| 43 | - | Kennung für Diagnosezähler (immer 10) |
| 44 | - | Länge (immer 14) |
| 45 | - | Formatcode (immer 0) |
| 46 | - | Zeitverzögerung vor Senden |
| 47 | 0 bis 7 | Netzknotenadresse |
| | 8 bis 15 | Reserviert |
| 48 | - | Zeitablauf zwischen Zeichen |
| 49 | - | RTS-Sendeverzögerung |
| 50 | - | RTS-Aus-Verzögerung |
| 51 | 0 bis 7 | Übertragungsgeschwindigkeit |
| | 8 bis 9 | Parität |
| | 10 bis 15 | Reserviert |
| 52 | - | Kennung für Diagnosezähler (immer 6) |
| 53 | - | Länge (immer 32) |
| 54 | - | Formatcode (immer 0) |
| 55 | - | Fehlercode Darstellungsschicht |
| 56 | - | Fehlerzähler Darstellungsschicht |
| 57 | - | Fehlercode Ausführungsfunktion |
| 58 | - | Letzter übertragener Ablaufunterbrechungscode |
| 59 | - | Datenfilenummer der Fehleraufforderung |
| 60 | - | Elementnummer der Fehleraufforderung |
| 61 | - | Funktionscode 1 Nachrichtenzähler |
| 62 | - | Funktionscode 2 Nachrichtenzähler |
| 63 | - | Funktionscode 3 Nachrichtenzähler |
| 64 | - | Funktionscode 4 Nachrichtenzähler |
| 65 | - | Funktionscode 5 Nachrichtenzähler |
| 66 | - | Funktionscode 6 Nachrichtenzähler |
| 67 | - | Funktionscode 8 Nachrichtenzähler |
| 68 | - | Funktionscode 15 Nachrichtenzähler |
| 69 | - | Funktionscode 16 Nachrichtenzähler |

## Ein-/Ausgangsstatusfile

Der E/A-Status-File (IOS) ist ein Nur-Lesen-File in der Steuerung, der Informationen zum Status der integrierten und lokalen Erweiterungs- E/A enthält. Der Datenfile hat folgende Struktur:

**Tabelle 3.19 E/A-Status-File**

| Wort | Beschreibung |
|---|---|
| 0 | Fehlercode für integrierte Module – immer 0 |
| 1 bis 6 | Fehlercode Erweiterungsmodule – Die Wortnummer entspricht der Steckplatznummer des Moduls. Weitere Informationen finden Sie in der Dokumentation des E/A-Moduls. *(MicroLogix 1200)* |
| 1 bis 16(1) | Fehlercode Erweiterungsmodule – Die Wortnummer entspricht der Steckplatznummer des Moduls. Weitere Informationen finden Sie in der Dokumentation des E/A-Moduls. *(MicroLogix 1500)* |

(1) 1 bis 8 für Basisgeräte der Serie A.

# Programmierbefehle – Übersicht

## Befehlssatz

In der folgenden Tabelle sind die Programmierbefehle für MicroLogix 1200- und MicroLogix 1500-Steuerungen innerhalb ihrer Funktionsgruppen dargestellt.(1)

| Funktionsgruppe | Beschreibung | Seite |
|---|---|---|
| Hochgeschwindigkeitszähler | HSL, RAC – Mit den Befehlen für Hochgeschwindigkeitszähler (und dem HSC-Funktionsfile) können Sie die Hochgeschwindigkeitsausgänge steuern und überwachen. Normalerweise mit DC-Eingängen verwendet. | 5-1 |
| Hochgeschwindigkeitsausgänge | PTO, PWM – Mit den Befehlen für Hochgeschwindigkeitsausgänge (sowie dem PTO- und dem PWM-Funktionsfile) können Sie die Hochgeschwindigkeitsausgänge steuern und überwachen. Normalerweise mit FET-Ausgängen (BXB-Geräte) verwendet. | 6-1 |
| Relais (Bit) | XIC, XIO, OTE, OTL, OTU, OSR, ONS, OSF – Die Relais- oder Bitbefehle überwachen und steuern den Status von Bits. | 7-1 |
| Zeitwerk und Zähler | TON, TOF, RTO, CTU, CTD, RES – Die Zeitwerk- und Zählerbefehle steuern zeitabhängige Vorgänge oder Vorgänge, die von der Anzahl von Ereignissen abhängen. | 8-1 |
| Compare | EQU, NEQ, LES, LEQ, GRT, GEQ, MEQ, LIM – Die Vergleichsbefehle vergleichen Werte mithilfe bestimmter Vergleichsoperationen. | 9-1 |
| Mathematische Befehle | ADD, SUB, MUL, DIV, NEG, CLR, ABS, SQR, SCL, SCP, SWP – Die mathematischen Befehle führen mathematische Operationen durch. | 10-1 |
| Konvertierung | DCD, ENC, TOD, FRD, GCD – Die Konvertierungsbefehle führen Multiplex- und Demultiplexoperationen mit Daten sowie Konvertierungen zwischen Binär- und Dezimalwerten aus. | 11-1 |
| Logik | AND, OR, XOR, NOT – Die Logikbefehle führen auf Bitbasis logische Operationen mit Worten aus. | 12-1 |
| Übertragung | MOV, MVM – Die Verschiebebefehle ändern und verschieben Worte. | 13-1 |
| File | CPW, COP, FLL, BSL, BSR, FFL, FFU, LFL, LFU – Die Filebefehle führen Operationen an Filedaten aus. | 14-1 |
| Schrittschaltwerk | SQC, SQO, SQL – Die Schrittschaltwerksbefehle steuern automatische Fertigungsmaschinen, die sich ständig wiederholende Aufgaben ausführen. | 15-1 |
| Programmsteuerung | JMP, LBL, JSR, SBR, RET, SUS, TND, MCR, END – Die Programmsteuerungsbefehle ändern den Ablauf der Programmausführung des Kontaktplans. | 16-1 |
| Eingang und Ausgang | IIM, IOM, REF – Mit den Eingangs- und Ausgangsbefehlen können Sie Daten gezielt und ohne Unterstützung durch die Eingangs- und Ausgangsabfragen aktualisieren. | 17-1 |
| Anwender-Interrupt | STS, INT, UID, UIE, UIF – Mit den Anwender-Interrupt-Befehlen können Sie Programm-Interrupts auf der Grundlage definierter Ereignisse erzeugen. | 18-1 |
| Prozesssteuerung | PID – Die Prozesssteuerungsbefehle ermöglichen eine Steuerung mit Rückführung. | 19-1 |
| ASCII | ABL, ACB, ACI, ACL, ACN, AEX, AHL, AIC, ARD, ARL, ASC, ASR, AWA, AWT – Mit den ASCII-Befehlen werden ASCII-Zeichenketten konvertiert und erstellt. Diese können nicht mit MicroLogix 1500 1764-LSP-Prozessoren der Serie A verwendet werden. | 20-1 |
| Kommunikation | MSG, SVC – Die Kommunikationsbefehle lesen Daten aus einer und schreiben Daten in eine andere Station. | 21-1 |
| Rezept *(nur MicroLogix 1500)* | RCP – Mit dem Rezeptbefehl können Sie einen Datensatz von der Rezept-Datenbank an einen Satz mit benutzerdefinierten Datentabellenelementen übertragen. | 22-1 |
| Datenprotokollierung *(nur MicroLogix 1500 1764-LRP)* | DLG – Der Datenprotokollierungsbefehl ermöglicht das Erfassen von Daten mit Zeit- und Datumsstempel. | 22-1 |

(1) Der „RTA-Befehl (Echtzeituhr anpassen)" wird auf Seite 3-5 nach den Informationen zu „Echtzeituhr-Funktionsfile" beschrieben.

## Befehlsbeschreibungen verwenden

In dem vorliegenden Handbuch wird für jeden Befehl (oder für jede Gruppe ähnlicher Befehle) eine Tabelle ähnlich der nachfolgenden verwendet. Diese Tabelle enthält Informationen zu allen Unterelementen (oder Komponenten) eines Befehls oder einer Befehlsgruppe. In dieser Tabelle wird der kompatible Adresstyp aufgeführt, der für jedes Unterelement eines Befehls oder einer Befehlsgruppe in einem Daten- oder Funktionsfile verwendet werden kann. Im Anschluss an diese Beispieltabelle finden Sie Definitionen der in diesen Tabellen verwendeten Begriffe.

**Tabelle 4.1 Gültige Adressierungsmodi und Filetypen - Beispieltabelle**

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DATI | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Source (Quelle) A | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Source (Quelle) B | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Ziel | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | | • | | | | | | • | • | | • | • | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig“) zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

Die in dieser Tabelle verwendeten Begriffe sind wie folgt definiert:

- Parameter - Der Parameter ist die Information, die dem Befehl durch den Anwender zur Verfügung gestellt wird. Dabei kann es sich um eine Adresse, einen Wert oder einen befehlsspezifischen Parameter wie eine Zeitbasis handeln.
- Datenfiles - Siehe „Datenfiles“ auf Seite 2-7.
- Funktionsfiles - Siehe „Funktionsfiles“ auf Seite 3-1.
- CS - Siehe „Kommunikations-Status-File“ auf Seite 3-14.
- IOS - Siehe „Ein-/Ausgangsstatusfile“ auf Seite 3-19.
- DLS - Siehe „Datenprotokoll-Statusfile“ auf Seite 22-14.
- Adressmodus - Siehe „Adressierungsmodi“ auf Seite 4-3.
- Adressierungsebene - Adressierungsebenen beschreiben die Speichereinheiten, auf die ein Operand eines Befehls angewendet werden kann. Relaistypbefehle (XIC, XIO usw.) müssen beispiels- weise auf Bitebene programmiert werden, Zeitwerkbefehle (TON, TOF usw.) auf Elementebene (Zeitwerke verfügen über 3 Wörter pro Element), und mathematische Befehle (ADD, SUB usw.) müssen auf Wort- oder Doppelwortebene programmiert werden.

## Adressierungsmodi

MicroLogix 1200- und MicroLogix 1500-Steuerungen unterstützen drei Arten der Datenadressierung:

- unmittelbar
- direkt
- indirekt

Die indizierte Adressierung wird durch MicroLogix 1200- und MicroLogix 1500-Steuerungen nicht unterstützt. Die indizierte Adressierung kann durch die indirekte Adressierung nachgestellt werden. Siehe „Beispiel – Verwenden indirekter Adressierung zum Duplizieren indizierter Adressierung“ auf Seite 4-7.

Die Art und Weise sowie der Zeitpunkt der Verwendung einer Adressierungsart hängen von dem zu programmierenden Befehl sowie dem Elementetyp ab, der in den Operanden des Befehls angegeben wird. Aufgrund der Unterstützung dieser drei Adressierungsmethoden bieten die MicroLogix 1200- und MicroLogix 1500-Steuerungen ein hohes Maß an Flexibilität bei der Überwachung oder Bearbeitung von Daten. Die drei Adressierungsmodi werden nachfolgend beschrieben.

### *Unmittelbare Adressierung*

Die unmittelbare Adressierung wird vor allem für die Zuordnung von numerischen Konstanten innerhalb von Befehlen verwendet. Beispiel: Sie benötigen ein 10-Sekunden-Zeitwerk. Hierfür programmieren Sie ein Zeitwerk mit der Zeitbasis 1 Sekunde und dem Sollwert 10. Die in diesem Beispiel verwendeten Zahlen 1 und 10 stellen jeweils eine unmittelbare Adressierung dar.

### *Direkte Adressierung*

Bei der direkten Adressierung wird ein bestimmter Datenstandort innerhalb der Steuerung definiert. Dabei kann jeder beliebige Datenstandort verwendet werden, der von den Elementen eines Operanden des zu programmierenden Befehls unterstützt wird. Beispiel eines Grenzwertbefehls:

- Unterer Grenzwert = Numerischer Wert (von -32768 bis 32767) über die Programmiersoftware eingegeben.
- Testwert = TPI:0.POT0 (Dies ist die aktuelle Position/der aktuelle Wert von Einstellpotentiometer 0.)
- Oberer Grenzwert = N7:17 (Dies sind die in Ganzzahl-File 7, Element 17 residenten Daten.)

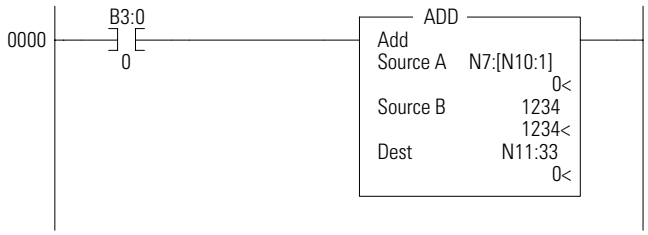
Der Testwert (TPI:0.POT0) und die Obergrenze (N7:17) sind Beispiele für die direkte Adressierung. Bei der Untergrenze wird die unmittel- bare Adressierung verwendet.

### *Indirekte Adressierung*

Bei der indirekten Adressierung können Komponenten der Adresse als Zeiger für andere Datenstandorte innerhalb der Steuerung verwendet werden. Diese Funktion kann vor allem bei bestimmten Anwendungstypen, beim Rezeptmanagement, der Batchverarbeitung usw., hilfreich sein. Allerdings ist die Fehlersuche und -behebung bei der indirekten Adressierung nicht immer einfach. Deshalb sollte die indirekte Adressierung nur verwendet werden, wenn dies für die zu entwickelnde Anwendung erforderlich ist.

Die MicroLogix 1200- und MicroLogix 1500-Steuerungen unterstützen die indirekte Adressierung von Files, Worten und Bits. Die Kompo- nenten einer Adresse, für die die indirekte Adressierung gilt, werden in eckige Klammern „[ ]“ gestellt. Die Verwendung der indirekten Adressierung wird anhand der folgenden Beispiele erläutert.

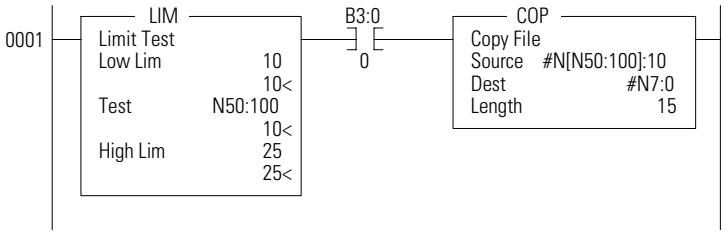
**Indirekte Adressierung eines Worts**




- Adresse: N7:[N10:1]
- In diesem Beispiel wird die Elementnummer, die für Quelle A in dem ADD-Befehl verwendet werden soll, durch die Nummer definiert, die sich am Standort N10:1 befindet. Ist der Wert des Datenstandorts N10:1 = 15, wird der ADD-Befehl als „N7:15 + Quelle B“ ausgeführt.
- In diesem Beispiel muss das durch N10:1 bezeichnete Element zwischen 0 und 255 liegen, da für alle Datenfiles eine maximale Größe von 256 Elementen gilt.

**HINWEIS** Wenn in N10:1 eine Zahl steht, die größer ist als die Anzahl der Elemente in dem Datenfile (wie in diesem Beispiel), kann die Datensicherheit nicht mehr garan- tiert werden, da Filegrenzen überschritten werden. Dadurch wird nicht unbedingt ein Steuerungsfehler erzeugt, doch der Datenstandort ist ungültig/unbekannt.

**Indirekte Adressierung eines Files**

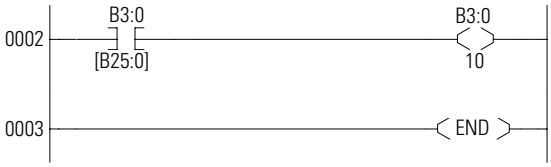



- Adresse: N[N50:100]:10
- Beschreibung: In diesem Beispiel wird die Quelle des COP-Befehls über N50:100 umgeleitet. Die Daten in N50:100 definieren die im Befehl zu verwendende Datenfile-Nummer. In diesem Beispiel wird die Quelle A für einen Kopierbefehl durch N[N50:100]:10 festgelegt. Bei der Abfrage des Befehls wird anhand der Daten am Standort N50:100 der Datenfile ermittelt, der für den COP-Befehl verwendet werden soll. Ist der Wert des Standorts N50:100 = 27, werden mit diesem Befehl 15 Elemente der Daten von N27:10 (N27:10 nach N27:24) nach N7:0 (N7:0 nach N7:14) kopiert.

**HINWEIS** Ist die unter N50:100 gespeicherte Zahl größer als 255 (wie in diesem Beispiel), wird ein Steuerungs- fehler erzeugt, da die Steuerung maximal 255 Datenfiles enthält. Außerdem sollte der File, der durch die indirekte Adressierung definiert wird, dem Filetyp entsprechen, der durch den Befehl definiert wird (in diesem Beispiel ein Ganzzahl-File).

**HINWEIS** Dieses Beispiel zeigt auch, wie eine Grenzwert- prüfung für die indirekte Adresse durchgeführt werden kann. Der Grenzwertbefehl am Anfang des Strompfads überwacht das indirekte Element. Wenn der Wert unter N50:100 kleiner als 10 oder größer als 25 ist, wird der Kopierbefehl nicht verarbeitet. Anhand dieser Vorgehensweise lässt sich sicherstellen, dass eine indirekte Adresse nicht irrtümlich auf Daten an einem anderen Standort zugreift.

**Indirekte Adressierung eines Bits**




- Adresse: B3/[B25:0]
- Beschreibung: In diesem Beispiel ist B25:0 das Element, für das die indirekte Adressierung gilt. Die unter B25:0 gespeicherten Daten bezeichnen das Bit innerhalb des Files B3. Ist der Wert des Standorts B25:0 = 1017, wird der XIC-Befehl mit B3/1017 verarbeitet.

**HINWEIS** Wenn in B25:0 eine Zahl steht, die größer als 4096 (oder die Anzahl der Elemente in dem Datenfile) ist (wie in diesem Beispiel), kann die Datensicher- heit nicht mehr garantiert werden. Bei Zahlen, die größer sind als die Anzahl der Elemente in dem Datenfile, wird die Filegrenze überschritten.

Dies sind nur einige Beispiele, die verwendet werden können. Weitere Beispiele sind:

- Indirekte Adressierung eines Files und Elements: N[N10:0]:[N25:0]
- Indirekte Adressierung eines Eingangssteckplatzes: I1:[N7:0].0

Die indirekte Adressierung ist nicht bei allen Befehlsgruppen möglich. Anhand der Kompatibilitätstabelle zu jedem Befehl können Sie feststellen, welche Elemente eines Befehls die indirekte Adressierung unterstützen.

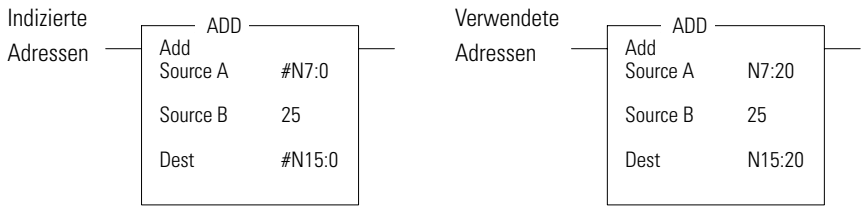
**WICHTIG** Gehen Sie bei der Verwendung der indirekten Adressierung äußerst sorgfältig vor. Bedenken Sie immer, dass unbeabsichtigt Filegrenzen überschritten oder auf falsche Daten verwiesen werden könnte.

## Beispiel – Verwenden indirekter Adressierung zum Duplizieren indizierter Adressierung

In diesem Abschnitt wird zunächst ein Beispiel der indizierten Adressierung gezeigt. Danach folgt ein Beispiel der entsprechenden indirekten Adressierung. Die indizierte Adressierung wird von speicherprogrammierbaren Steuerungen SLC 500 und MicroLogix 1000 unterstützt. Die indizierte Adressierung wird durch MicroLogix 1200- und MicroLogix 1500-Steuerungen nicht unterstützt. Das vorliegende Beispiel wird zu Vergleichszwecken präsentiert.

### *Beispiel für indizierte Adressierung*

Der folgende ADD-Befehl verwendete für die Adressen von Quelle A und Ziel die indizierte Adressierung. Lautet der indizierte Offset-Wert 20 (gespeichert in S:24), verwendet die Steuerung die an der Basisadresse gespeicherten Daten plus den indizierten Offset zum Durchführen der Operation.




In dem vorliegenden Beispiel verwendet die Steuerung die folgende Adresse:

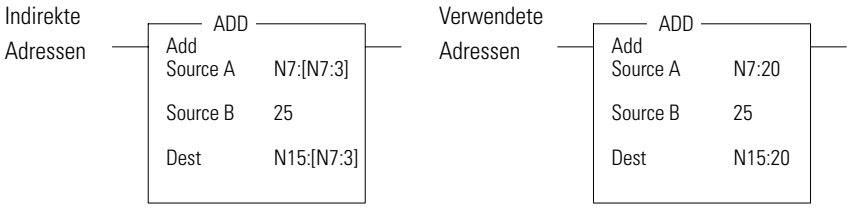
| Operand | Basisadresse | Offset-Wert in S:24 | Verwendete Adresse |
|---|---|---|---|
| Source (Quelle) A | N7:0 | 20 | N7:20 |
| Ziel | N15:0 | 20 | N15:20 |

**HINWEIS** In den SLC- und ML1000-Steuerungen stehen einige Befehle zur Verfügung, die S:24 nach Abschluss des Befehls löschen. Aus diesem Grund sollten Sie vor der Ausführung eines indizierten Befehls sicherstellen, dass das Indexregister den gewünschten Wert enthält.

### *Beispiel für indirekte Adressierung*

Nachfolgend sehen Sie ein Beispiel der entsprechenden indirekten Adressierung. Anstatt des Indexregisters S:24 können Sie eine beliebige gültige Wortadresse als indirekte Adresse angeben. Dabei können innerhalb eines Befehls mehrere indirekte Adressen ver- wendet werden.

Der folgende ADD-Befehl verwendet für die Adressen von Quelle A und Ziel die indirekte Adressierung. Wenn der indirekte Offset-Wert gleich 20 ist (gespeichert in N7:3), verwendet die Steuerung zur Durchführung des Befehls die an der Basisadresse gespeicherten Daten plus den indirekten Offset.




In dem vorliegenden Beispiel verwendet die Steuerung die folgende Adresse:

| Operand | Basisadresse | Offset-Wert in N7:3 | Verwendete Adresse |
|---|---|---|---|
| Source (Quelle) A | N7:0 | 20 | N7:20 |
| Ziel | N7:0 | 20 | N15:20 |

# Verwenden des Hochgeschwindigkeitszählers und des programmierbaren Endschalters

## Hochgeschwindigkeitszähler – Übersicht

Die MicroLogix 1200-Steuerung verfügt über einen 20-kHz-Hochgeschwindigkeitszähler, die MicroLogix 1500-Steuerung verfügt über zwei Zähler. Die Funktionsweise der Zähler ist identisch. Jeder Zähler weist vier dedizierte Eingänge auf, die gegen andere Eingänge auf der Steuerung isoliert sind. HSC0 verwendet die Eingänge 0 bis 3, und HSC1 (*nur MicroLogix 1500*) verwendet die Eingänge 4 bis 7. Die Zähler arbeiten unabhängig voneinander.

**HINWEIS** Die Funktionsweise des Hochgeschwindigkeitszählers wird in dem vorliegenden Dokument anhand des Hochgeschwindigkeitszählers HSC0 erläutert. Der Hochgeschwindigkeitszähler HSC1 der MicroLogix 1500-Steuerung weist dieselbe Funktionalität auf.

**WICHTIG** Die HSC-Funktion kann nur mit den eingebetteten E/A der Steuerung verwendet werden. Sie kann nicht mit Erweiterungs-E/A-Modulen verwendet werden.

In diesem Kapitel werden die Funktionsweise des Hochgeschwindigkeitszählers sowie die Befehle HSL und RAC unter folgenden Über-schriften beschrieben:


- „HSC-Funktionsfile (Hochgeschwindigkeitszähler)“ auf Seite 5-2.
- „HSL – Hochgeschwindigkeitszähler laden“ auf Seite 5-26.
- „RAC – Istwert zurücksetzen“ auf Seite 5-27.

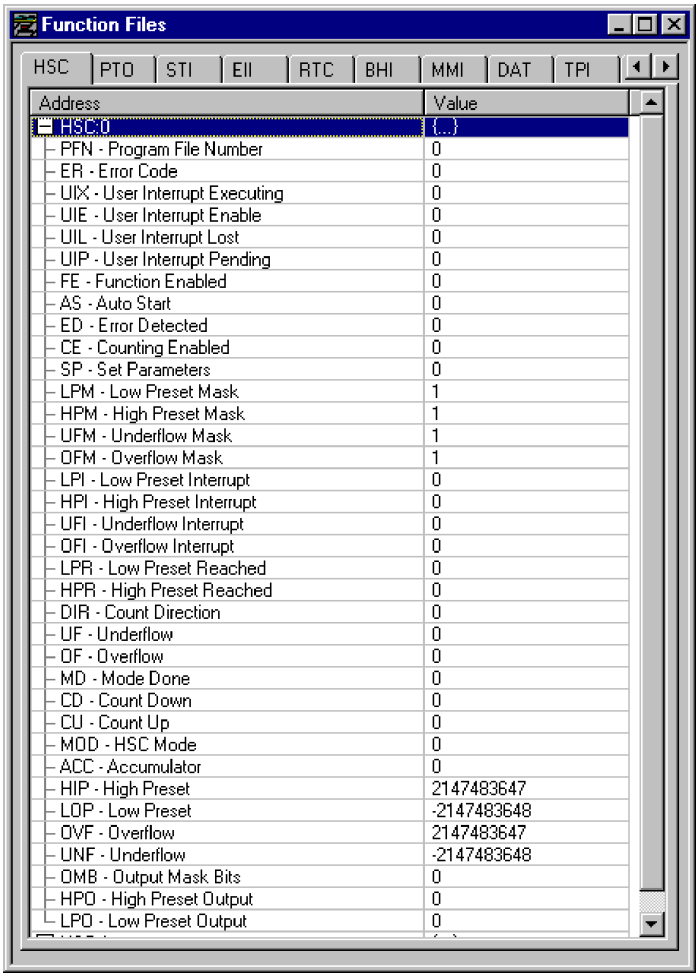

## Programmierbarer Endschalter – Übersicht

Mit der Funktion für programmierbare Endschalter können Sie den Hochgeschwindigkeitszähler so konfigurieren, dass er als programmierbarer Endschalter (PLS) oder als Nockendrehschalter arbeitet. Weitere Informationen finden Sie auf Seite 5-28.

## HSC-Funktionsfile (Hochgeschwindigkeits-zähler)

In dem RSLogix 500-Funktionsfileordner finden Sie einen HSC-Funktionsfile. Dieser File ermöglicht den Zugriff auf HSC-Konfigurationsdaten; außerdem kann das Steuerungsprogramm über diesen File auf alle Informationen zu den Hochgeschwindigkeits-zählern zugreifen.

**HINWEIS** Wenn sich die Steuerung im Run-Modus befindet, ändern sich unter Umständen die Daten in den Unterelement-Feldern.

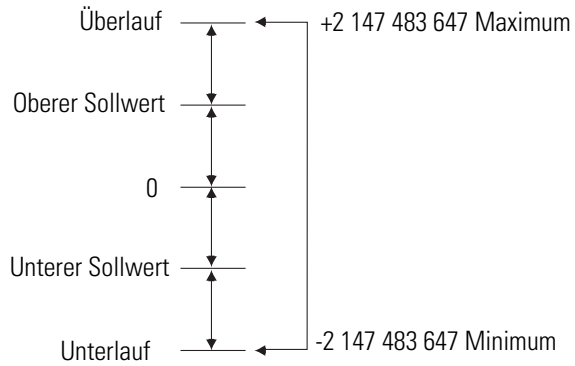

Function Files
HSC
PTO
STI
EII
RTC
BHI
MMI
DAT
TPI
Address
Value
HSC:0 {...}
PFN - Program File Number 0
ER - Error Code 0
UIX - User Interrupt Executing 0
UIE - User Interrupt Enable 0
UIL - User Interrupt Lost 0
UIP - User Interrupt Pending 0
FE - Function Enabled 0
AS - Auto Start 0
ED - Error Detected 0
CE - Counting Enabled 0
SP - Set Parameters 0
LPM - Low Preset Mask 1
HPM - High Preset Mask 1
UFM - Underflow Mask 1
OFM - Overflow Mask 1
LPI - Low Preset Interrupt 0
HPI - High Preset Interrupt 0
UFI - Underflow Interrupt 0
OFI - Overflow Interrupt 0
LPR - Low Preset Reached 0
HPR - High Preset Reached 0
DIR - Count Direction 0
UF - Underflow 0
OF - Overflow 0
MD - Mode Done 0
CD - Count Down 0
CU - Count Up 0
MOD - HSC Mode 0
ACC - Accumulator 0
HIP - High Preset 2147483647
LOP - Low Preset -2147483648
OVF - Overflow 2147483647
UNF - Underflow -2147483648
OMB - Output Mask Bits 0
HPO - High Preset Output 0
LPO - Low Preset Output 0

Die HSC-Funktion sowie die Befehle PTO und PWM unterscheiden sich von den meisten anderen Steuerungsbefehlen. Diese Befehle werden von speziellen Schaltungen ausgeführt, die parallel zum Hauptsystemprozessor aktiv sind. Dieser Aufbau ist aufgrund der Hochleistungsanforderungen dieser Funktionen erforderlich.

Der Hochgeschwindigkeitszähler ist äußerst vielseitig. An beiden Hochgeschwindigkeitszählern kann einer der insgesamt acht Betriebsmodi ausgewählt oder konfiguriert werden. (Die Betriebsmodi werden an späterer Stelle in diesem Kapitel erläutert. Siehe den Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16). Zu den erweiterten Leistungsmerkmalen der Hochgeschwindigkeitszähler zählen unter anderem:

- 20-kHz-Betrieb
- Hochgeschwindigkeits-Direktsteuerung der Ausgänge
- Ganzzahlige 32-Bit-Daten mit Vorzeichen (Zählbereich von ± 2 147 483 647)
- Programmierbare obere und untere Sollwerte und Überlauf- und Unterlaufsollwerte
- automatische Interrupt-Verarbeitung aufgrund Istwert-Zählung
- Bearbeitung von Parametern während Ausführung (über Anwendersteuerprogramm)

Das folgende Diagramm zeigt die Funktionsweise des Hochgeschwindigkeitszählers.



## Zusammenfassung der Unterelemente des HSC-Files

Jeder HSC-File besteht aus 36 Unterelementen. Bei diesen Unter-elementen handelt es sich um Bits, Worte oder Doppelworte, die eine Steuerung der HSC-Funktion ermöglichen oder HSC-Statusinformationen für das Steuerprogramm bereitstellen. Die einzelnen Unterelemente und ihre Funktionen werden in diesem Kapitel beschrieben. Die unten stehende Tabelle enthält eine Zusammenfassung der Unter-elemente. Die Beispiele beziehen sich immer auf den Hochgeschwin-digkeitszähler HSC0, doch der Hochgeschwindigkeits-zähler HSC1 weist exakt dieselben Begriffe und dasselbe Verhalten auf.

**Tabelle 5.1 HSC-Funktionsfile (HSC:0 oder HSC:1)**

| Beschreibung Unterelement | Adresse | Datenformat | HSC-Modi(1) | Funktion | Anwenderpro-grammzugriff | Weitere Informationen |
|---|---|---|---|---|---|---|
| PFN - Programmfilenummer | HSC:0.PFN | Wort (INT) | 0 bis 7 | Steuerung | Nur Lesen | 5-5 |
| ER - Fehlercode | HSC:0.ER | Wort (INT) | 0 bis 7 | Status | Nur Lesen | 5-5 |
| UIX - Anwender-Interrupt-Ausführung | HSC:0/UIX | Bit | 0 bis 7 | Status | Nur Lesen | 5-8 |
| UIE - Anwender-Interrupt aktivieren | HSC:0/UIE | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-8 |
| UIL - Anwender-Interrupt-Verlust | HSC:0/UIL | Bit | 0 bis 7 | Status | Lesen/Schreiben | 5-9 |
| UIP - Anwender-Interrupt anstehend | HSC:0/UIP | Bit | 0 bis 7 | Status | Nur Lesen | 5-9 |
| FE - Funktion aktiviert | HSC:0/FE | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-6 |
| AS - Auto-Start | HSC:0/AS | Bit | 0 bis 7 | Steuerung | Nur Lesen | 5-6 |
| ED - Fehler erkannt | HSC:0/ED | Bit | 0 bis 7 | Status | Nur Lesen | 5-6 |
| CE - Zählen aktiviert | HSC:0/CE | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-7 |
| SP - Parameter einstellen | HSC:0/SP | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-7 |
| LPM - Untere Sollwert-Maske | HSC:0/LPM | Bit | 2 bis 7 | Steuerung | Lesen/Schreiben | 5-9 |
| HPM - Obere Sollwert-Maske | HSC:0/HPM | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-11 |
| UFM - Unterlauf-Maske | HSC:0/UFM | Bit | 2 bis 7 | Steuerung | Lesen/Schreiben | 5-12 |
| OFM - Überlauf-Maske | HSC:0/OFM | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-14 |
| LPI - Unterer Sollwert erreicht, Interrupt | HSC:0/LPI | Bit | 2 bis 7 | Status | Lesen/Schreiben | 5-10 |
| HPI - Oberer Sollwert erreicht, Interrupt | HSC:0/HPI | Bit | 0 bis 7 | Status | Lesen/Schreiben | 5-11 |
| UFI - Interrupt durch Unterlauf | HSC:0/UFI | Bit | 2 bis 7 | Status | Lesen/Schreiben | 5-13 |
| OFI - Interrupt durch Überlauf | HSC:0/OFI | Bit | 0 bis 7 | Status | Lesen/Schreiben | 5-14 |
| LPR - Unterer Sollwert erreicht | HSC:0/LPR | Bit | 2 bis 7 | Status | Nur Lesen | 5-10 |
| HPR - Oberer Sollwert erreicht | HSC:0/HPR | Bit | 2 bis 7 | Status | Nur Lesen | 5-12 |
| DIR - Zählrichtung | HSC:0/DIR | Bit | 0 bis 7 | Status | Nur Lesen | 5-15 |
| UF - Unterlauf | HSC:0/UF | Bit | 0 bis 7 | Status | Lesen/Schreiben | 5-12 |
| OF - Überlauf | HSC:0/OF | Bit | 0 bis 7 | Status | Lesen/Schreiben | 5-13 |
| MD - Modus fertig | HSC:0/MD | Bit | 0 oder 1 | Status | Lesen/Schreiben | 5-15 |
| CD - Abwärtszähler | HSC:0/CD | Bit | 2 bis 7 | Status | Nur Lesen | 5-15 |
| CU - Aufwärtszähler | HSC:0/CU | Bit | 0 bis 7 | Status | Nur Lesen | 5-16 |
| MOD - HSC-Modus | HSC:0.MOD | Wort (INT) | 0 bis 7 | Steuerung | Nur Lesen | 5-16 |
| ACC - Istwert | HSC:0.ACC | Doppelwort (32-Bit INT) | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-22 |
| HIP - Oberer Sollwert | HSC:0.HIP | Doppelwort (32-Bit INT) | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-22 |
| LOP - Unterer Sollwert | HSC:0.LOP | Doppelwort (32-Bit INT) | 2 bis 7 | Steuerung | Lesen/Schreiben | 5-22 |
| OVF - Überlauf | HSC:0.OVF | Doppelwort (32-Bit INT) | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-23 |
| UNF - Unterlauf | HSC:0.UNF | Doppelwort (32-Bit INT) | 2 bis 7 | Steuerung | Lesen/Schreiben | 5-23 |
| OMB - Ausgangsmaske | HSC:0.OMB | Wort (16-Bit-Binärwert) | 0 bis 7 | Steuerung | Nur Lesen | 5-24 |
| HPO - Ausgang bei oberem Sollwert | HSC:0.HPO | Wort (16-Bit-Binärwert) | 0 bis 7 | Steuerung | Lesen/Schreiben | 5-25 |
| LPO - Ausgang bei unterem Sollwert | HSC:0.LPO | Wort (16-Bit-Binärwert) | 2 bis 7 | Steuerung | Lesen/Schreiben | 5-25 |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16.
n/v = nicht vorhanden

## Unterelemente des HSC-Funktionsfiles

Die Beispiele beziehen sich immer auf den Hochgeschwindigkeits- zähler HSC0, doch der Hochgeschwindigkeitszähler HSC1 weist exakt dieselben Begriffe und dasselbe Verhalten auf.

### Programmfilenummer (PFN)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| PFN - Programmfilenummer | HSC:0.PFN | Wort (INT) | 0 bis 7 | Steuerung | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Mit der PFN-Variablen (Programmfilenummer) wird festgelegt, welches Unterprogramm aufgerufen (ausgeführt) wird, wenn die HSC0-Zählung den oberen oder unteren Sollwert erreicht oder den Über- oder Unterlaufwert überschreitet. Der ganzzahlige Wert dieser Variablen bezeichnet den Programmfile, der dann ausgeführt wird. Als Unterprogrammfile kann jeder Programmfile (3 bis 255) festgelegt werden.

Siehe auch: „Interrupt-Latenzzeit" auf Seite 18-5.

### Fehlercode (ER)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| ER - Fehlercode | HSC:0.ER | Wort (INT) | 0 bis 7 | Status | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Die Fehlercodes, die von dem HSC-Subsystem erkannt wurden, werden in diesem Wort angezeigt. Zu diesen Fehlern gehören:

**Tabelle 5.2 HSC-Fehlercodes**

| Fehlercode | Name | Modus(1) | Beschreibung |
|---|---|---|---|
| 1 | ungültige Filenummer | nicht vorhanden | Interrupt-(Programm-)File in HSC:0.PFN ist kleiner als 3, größer als 255 oder nicht vorhanden |
| 2 | ungültiger Modus | nicht vorhanden | ungültiger Modus(1) |
| 3 | ungültiger oberer Sollwert | 0,1 | Oberer Sollwert ist kleiner als oder gleich null (0) |
| | | 2 bis 7 | Oberer Sollwert ist kleiner als oder gleich unterem Sollwert |
| 4 | ungültiger Überlauf | 0 bis 7 | Oberer Sollwert ist größer als Überlauf |

(1) Beschreibungen der Modi finden Sie im Abschnit „HSC-Modus (MOD)" auf Seite 5-16.

## Funktion aktiviert (FE)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| FE - Funktion aktiviert | HSC:0/FE | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das FE-Bit (Funktion aktiviert) ist ein Status-/Steuer-Bit, das festlegt, wann der HSC-Interrupt aktiviert wird, und dass die von dem HSC generierten Interrupts entsprechend ihrer Priorität verarbeitet werden.

Dieses Bit kann über das Anwenderprogramm gesetzt/gelöscht werden; es wird automatisch durch das HSC-Subsystem gesetzt, wenn Auto-Start aktiviert wurde.

Siehe auch: „Priorität bei Anwender-Interrupts" auf Seite 18-4.

## Auto-Start (AS)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| AS - Auto-Start | HSC:0/AS | Bit | 0 bis 7 | Steuerung | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das Auto-Start-Bit (AS) wird über das Programmiergerät konfiguriert und als Teil des Anwenderprogramms gespeichert. Das Auto-Start-Bit legt fest, ob die HSC-Funktion automatisch gestartet wird, sobald ein Run- oder Testmodus der Steuerung aktiviert wird. Außerdem muss zur Aktivierung des Hochgeschwindigkeitszählers das CE-Bit (Zählen aktiviert) gesetzt sein.

## Fehler erkannt (ED)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| ED - Fehler erkannt | HSC:0/ED | Bit | 0 bis 7 | Status | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das ED-Flag (Fehler erkannt) ist ein Status-Bit, das in dem Steuer- programm verwendet werden kann, um Fehler in dem HSC-Sub- system festzustellen. Die häufigsten Fehler, die dieses Bit darstellt, sind Konfigurationsfehler. Wenn dieses Bit gesetzt ist (1), sollte der spezifische Fehlercode in dem Parameter HSC:0.ER genauer geprüft werden.

Dieses Bit wird durch die Steuerung verwaltet und automatisch gesetzt und gelöscht.

## Zählen aktiviert (CE)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| CE - Zählen aktiviert | HSC:0/CE | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Mit dem CE-Steuer-Bit (Zählen aktiviert) wird der Hochgeschwindigkeitszähler aktiviert oder deaktiviert. Durch Setzen dieses Bits (1) wird die Zählung aktiviert; wenn das Bit gelöscht ist (0, Standard), ist die Zählung deaktiviert. Wenn dieses Bit deaktiviert wird, während der Zähler läuft, wird der Istwert gehalten. Sobald das Bit wieder gesetzt ist, wird die Zählung fortgesetzt.

Dieses Bit kann durch das Anwenderprogramm gesteuert werden; der Wert dieses Bits bleibt auch nach Aus- und Einschalten der Spannungsversorgung erhalten. Dieses Bit muss zur Aktivierung des Hochgeschwindigkeitszählers gesetzt werden.

## Parameter einstellen (SP)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| SP - Parameter einstellen | HSC:0/SP | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Mit dem SP-Steuer-Bit (Parameter einstellen) werden neue Variablen in das HSC-Subsystem geladen. Wenn ein OTE-Befehl mit der Adresse HSC:0/SP wahr ist (Strompfadübergang aus-ein), werden alle aktuell in der HSC-Funktion gespeicherten Konfigurationsvariablen geprüft und in das HSC-Subsystem geladen. Das HSC-Subsystem arbeitet dann mit den neu geladenen Einstellungen.

Dieses Bit wird durch das Anwenderprogramm gesteuert; der Wert dieses Bits bleibt auch nach Aus- und Einschalten der Spannungsver- sorgung erhalten. Dieses Bit wird durch das Anwenderprogramm gesetzt und gelöscht. Das SP-Bit kann während der HSC-Ausführung ohne Zählerverlust umgeschaltet werden.

## Anwender-Interrupt aktivieren (UIE)

| Beschreibung | Adresse | Daten-format | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| UIE - Anwender-Interrupt aktivieren | HSC:0/UIE | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Mit dem UIE-Bit (Anwender-Interrupt aktivieren) wird die Verarbeitung von HSC-Unterprogrammen aktiviert oder deaktiviert. Dieses Bit muss gesetzt werden (1), wenn das HSC-Unterprogramm durch die Steuerung verarbeitet werden soll, sobald eine der folgenden Bedingungen eintritt:

- Unterer Sollwert erreicht
- Oberer Sollwert erreicht
- Überlaufwert erreicht - über Überlaufwert hinaus aufwärts zählen
- Unterlaufwert erreicht - über Unterlaufwert hinaus abwärts zählen

Wenn dieses Bit gelöscht wird (0), erfolgt keine automatische Abfrage der HSC-Unterprogramme durch das HSC-Subsystem. Dieses Bit kann über das Anwenderprogramm gesteuert werden (mit den Befehlen OTE, UIE oder UID).

**ACHTUNG**

Wenn Interrupts während der Programmabfrage über einen OTL-, OTE- oder UIE-Befehl aktiviert werden, *muss* dieser Befehl der *letzte* Befehl sein, der auf dem Strompfad ausgeführt wird (letzter Befehl auf letztem Abzweig). Es wird empfohlen, in dem Strompfad diesen Befehl als einzigen Ausgangsbefehl zu verwenden.

## Anwender-Interrupt-Ausführung (UIX)

| Beschreibung | Adresse | Daten-format | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| UIX - Anwender-Interrupt-Ausführung | HSC:0/UIX | Bit | 0 bis 7 | Status | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das UIX-Bit (Anwender-Interrupt-Ausführung) wird gesetzt (1), sobald das HSC-Subsystem die Verarbeitung der HSC-Unterprogramme aufgrund einer der folgenden Bedingungen aufnimmt:

- Unterer Sollwert erreicht
- Oberer Sollwert erreicht
- Überlaufwert erreicht - über Überlaufwert hinaus aufwärts zählen
- Unterlaufwert erreicht - über Unterlaufwert hinaus abwärts zählen

Das UIX-Bit kann in dem Steuerprogramm als bedingte Logik eingesetzt werden, um festzustellen, ob ein HSC-Interrupt gerade ausgeführt wird.

Das HSC-Subsystem löscht (0) das UIX-Bit, wenn die Verarbeitung der HSC-Unterprogramme durch die Steuerung abgeschlossen ist.

## Anwender-Interrupt anstehend (UIP)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| UIP - Anwender-Interrupt anstehend | HSC:0/UIP | Bit | 0 bis 7 | Status | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16.

Das UIP-Status-Flag (Anwender-Interrupt anstehend) zeigt an, dass ein Interrupt ansteht. Dieses Status-Bit kann überwacht oder als Logik in dem Steuerprogramm eingesetzt werden, um festzustellen, ob ein Unterprogramm sofort ausgeführt werden kann.

Dieses Bit wird durch die Steuerung verwaltet und automatisch gesetzt und gelöscht.

## Anwender-Interrupt-Verlust (UIL)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| UIL - Anwender-Interrupt-Verlust | HSC:0/UIL | Bit | 0 bis 7 | Status | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16.

Das UIL-Status-Flag (Anwender-Interrupt-Verlust) zeigt an, dass ein Interrupt verloren wurde. Die Steuerung kann eine aktive Anwender- Interrupt-Bedingung verarbeiten und bis zu zwei weitere, anstehende Interrupt-Bedingungen speichern.

Dieses Bit wird von der Steuerung gesetzt. Die Verlustbedingung muss durch das Steuerprogramm verwendet, ggf. verfolgt und gelöscht werden.

## Untere Sollwert-Maske (LPM)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| LPM - Untere Sollwert-Maske | HSC:0/LPM | Bit | 2 bis 7 | Steuerung | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16.

Mit dem LPM-Steuer-Bit (Untere Sollwert-Maske) wird die Möglichkeit eines Interrupts bei Erreichen des unteren Sollwerts aktiviert (Interrupt möglich) oder deaktiviert (Interrupt nicht möglich). Wenn dieses Bit gelöscht ist (0) und die Bedingung „Unterer Sollwert erreicht“ durch den HSC erkannt wird, wird der HSC-Anwender-Interrupt nicht ausgeführt.

Dieses Bit wird durch das Anwenderprogramm gesteuert; der Wert dieses Bits bleibt auch nach Aus- und Einschalten der Spannungsver- sorgung erhalten. Dieses Bit wird durch das Anwenderprogramm gesetzt und gelöscht.

## Unterer Sollwert erreicht, Interrupt (LPI)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| LPI - Unterer Sollwert erreicht, Interrupt | HSC:0/LPI | Bit | 2 bis 7 | Status | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16.

Das LPI-Status-Bit (Unterer Sollwert erreicht, Interrupt) wird gesetzt (1), wenn der HSC-Istwert den unteren Sollwert erreicht und der HSC-Interrupt ausgelöst wurde. Mit diesem Bit kann in dem Steuerpro-gramm angezeigt werden, dass der HSC-Interrupt durch Erreichen des unteren Sollwerts ausgelöst wurde. Wenn das Steuerprogramm nach Erreichen des unteren Sollwerts eine bestimmte Steueraktion durchführen soll, kann dieses Bit als bedingte Logik verwendet werden.

Dieses Bit kann durch das Steuerprogramm gelöscht (0) werden; es wird außerdem durch das HSC-Subsystem gelöscht, sobald folgende Bedingungen eintreten:

- Ausführung eines Interrupts nach Erreichen des oberen Sollwerts
- Ausführung eines Interrupts nach Erreichen des Unterlaufwerts
- Ausführung eines Interrupts nach Erreichen des Überlaufwerts
- Aktivierung des Ausführungsmodus der Steuerung

## Unterer Sollwert erreicht (LPR)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| LPR - Unterer Sollwert erreicht | HSC:0/LPR | Bit | 2 bis 7 | Status | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16.

Das LPR-Status-Flag (Unterer Sollwert erreicht) wird durch das HSC-Subsystem gesetzt (1), sobald der Istwert (HSC:0.ACC) kleiner als oder gleich der Variablen für den unteren Sollwert (HSC:0.LOP) ist.

Dieses Bit wird durch das HSC-Subsystem kontinuierlich aktualisiert, wenn sich die Steuerung in einem Ausführungsmodus befindet.

## Obere Sollwert-Maske (HPM)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| HPM - Obere Sollwert-Maske | HSC:0/HPM | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Mit dem HPM-Steuer-Bit (Obere Sollwert-Maske) wird die Möglichkeit eines Interrupts bei Erreichen des oberen Sollwerts aktiviert (Interrupt möglich) oder deaktiviert (Interrupt nicht möglich). Wenn dieses Bit gelöscht ist (0) und die Bedingung „Oberer Sollwert erreicht" durch den HSC erkannt wird, wird der HSC-Anwender-Interrupt nicht ausgeführt.

Dieses Bit wird durch das Anwenderprogramm gesteuert; der Wert dieses Bits bleibt auch nach Aus- und Einschalten der Spannungs-versorgung erhalten. Dieses Bit wird durch das Anwenderprogramm gesetzt und gelöscht.

## Oberer Sollwert erreicht, Interrupt (HPI)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| HPI - Oberer Sollwert erreicht, Interrupt | HSC:0/HPI | Bit | 0 bis 7 | Status | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das HPI-Status-Bit (Oberer Sollwert erreicht, Interrupt) wird gesetzt (1), wenn der HSC-Istwert den oberen Sollwert erreicht und der HSC-Interrupt ausgelöst wurde. Mit diesem Bit kann in dem Steuer- programm angezeigt werden, dass der HSC-Interrupt durch Erreichen des oberen Sollwerts ausgelöst wurde. Wenn das Steuerprogramm nach Erreichen des oberen Sollwerts eine bestimmte Steueraktion durchführen soll, wird dieses Bit als bedingte Logik verwendet.

Dieses Bit kann durch das Steuerprogramm gelöscht (0) werden; es wird außerdem durch das HSC-Subsystem gelöscht, sobald folgende Bedingungen eintreten:

- Ausführung eines Interrupts nach Erreichen des unteren Sollwerts
- Ausführung eines Interrupts nach Erreichen des Unterlaufwerts
- Ausführung eines Interrupts nach Erreichen des Überlaufwerts
- Aktivierung des Ausführungsmodus der Steuerung

## Oberer Sollwert erreicht (HPR)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| HPR - Oberer Sollwert erreicht | HSC:0/HPR | Bit | 2 bis 7 | Status | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das HPR-Status-Flag (Oberer Sollwert erreicht) wird durch das HSC-Subsystem gesetzt (1), sobald der Istwert (HSC:0.ACC) größer als oder gleich der Variablen für den oberen Sollwert (HSC:0.HIP) ist.

Dieses Bit wird durch das HSC-Subsystem kontinuierlich aktualisiert, wenn sich die Steuerung in einem Ausführungsmodus befindet.

## Unterlauf (UF)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| UF - Unterlauf | HSC:0/UF | Bit | 0 bis 7 | Status | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das UF-Status-Flag (Unterlauf) wird durch das HSC-Subsystem gesetzt (1), sobald der Istwert (HSC:0.ACC) den Wert der Unterlaufvariablen (HSC:0.UNF) unterschritten hat.

Dieses Bit wird vorübergehend durch das HSC-Subsystem gesetzt.
Die Unterlaufbedingung muss dann durch das Steuerprogramm verwendet, ggf. verfolgt und gelöscht (0) werden.

Unterlaufbedingungen führen nicht zu einem Steuerungsfehler.

## Unterlauf-Maske (UFM)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| UFM - Unterlauf-Maske | HSC:0/UFM | Bit | 2 bis 7 | Steuerung | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Mit dem UFM-Steuer-Bit (Unterlauf-Maske) wird die Möglichkeit eines Interrupts bei Erreichen des Unterlaufwerts aktiviert (Interrupt möglich) oder deaktiviert (Interrupt nicht möglich). Wenn dieses Bit gelöscht ist (0) und die Bedingung „Unterlaufwert erreicht" durch den HSC erkannt wird, wird der HSC-Anwender-Interrupt nicht ausgeführt.

Dieses Bit wird durch das Anwenderprogramm gesteuert; der Wert dieses Bits bleibt auch nach Aus- und Einschalten der Spannungs- versorgung erhalten. Dieses Bit wird durch das Anwenderprogramm gesetzt und gelöscht.

## Interrupt durch Unterlauf (UFI)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| UFI - Interrupt durch Unterlauf | HSC:0/UFI | Bit | 2 bis 7 | Status | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16.

Das UFI-Status-Bit (Interrupt durch Unterlauf) wird gesetzt (1), wenn der HSC-Istwert den Unterlaufwert erreicht und der HSC-Interrupt ausgelöst wurde. Mit diesem Bit kann in dem Steuerprogramm angezeigt werden, dass der HSC-Interrupt durch Erreichen des Unterlaufwerts ausgelöst wurde. Wenn das Steuerprogramm nach Erreichen des Unterlaufwerts eine bestimmte Steueraktion durch-führen soll, wird dieses Bit als bedingte Logik verwendet.

Dieses Bit kann durch das Steuerprogramm gelöscht (0) werden; es wird außerdem durch das HSC-Subsystem gelöscht, sobald folgende Bedingungen eintreten:

- Ausführung eines Interrupts nach Erreichen des unteren Sollwerts
- Ausführung eines Interrupts nach Erreichen des oberen Sollwerts
- Ausführung eines Interrupts nach Erreichen des Überlaufwerts
- Aktivierung des Ausführungsmodus der Steuerung

## Überlauf (OF)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| OF - Überlauf | HSC:0/OF | Bit | 0 bis 7 | Status | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16.

Das OF-Status-Flag (Überlauf) wird durch das HSC-Subsystem gesetzt (1), sobald der Istwert (HSC:0.ACC) den Wert der Überlaufvariablen (HSC:0.OF) überschritten hat.

Dieses Bit wird vorübergehend durch das HSC-Subsystem gesetzt. Die Überlaufbedingung muss dann durch das Steuerprogramm verwendet, ggf. verfolgt und gelöscht (0) werden.

Überlaufbedingungen führen nicht zu einem Steuerungsfehler.

## Überlauf-Maske (OFM)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| OFM - Überlauf-Maske | HSC:0/OFM | Bit | 0 bis 7 | Steuerung | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Mit dem OFM-Steuer-Bit (Überlauf-Maske) wird die Möglichkeit eines Interrupts bei Erreichen des Überlaufwerts aktiviert (Interrupt möglich) oder deaktiviert (Interrupt nicht möglich). Wenn dieses Bit gelöscht ist (0) und die Bedingung „Überlaufwert erreicht" durch den HSC erkannt wird, wird der HSC-Anwender-Interrupt nicht ausgeführt.

Dieses Bit wird durch das Anwenderprogramm gesteuert; der Wert dieses Bits bleibt auch nach Aus- und Einschalten der Spannungsver- sorgung erhalten. Dieses Bit wird durch das Anwenderprogramm gesetzt und gelöscht.

## Interrupt durch Überlauf (OFI)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| OFI - Interrupt durch Überlauf | HSC:0/OFI | Bit | 0 bis 7 | Status | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das OFI-Status-Bit (Interrupt durch Überlauf) wird gesetzt (1), wenn der HSC-Istwert den Überlaufwert erreicht und der HSC-Interrupt ausgelöst wurde. Mit diesem Bit kann in dem Steuerprogramm angezeigt werden, dass der HSC-Interrupt durch die Überlaufvariable ausgelöst wurde. Wenn das Steuerprogramm nach Erreichen des Überlaufwerts eine bestimmte Steueraktion durchführen soll, wird dieses Bit als bedingte Logik verwendet.

Dieses Bit kann durch das Steuerprogramm gelöscht (0) werden; es wird außerdem durch das HSC-Subsystem gelöscht, sobald folgende Bedingungen eintreten:

- Ausführung eines Interrupts nach Erreichen des unteren Sollwerts
- Ausführung eines Interrupts nach Erreichen des oberen Sollwerts
- Ausführung eines Interrupts nach Erreichen des Unterlaufwerts
- Aktivierung des Ausführungsmodus der Steuerung

## Zählrichtung (DIR)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| DIR - Zählrichtung | HSC:0/DIR | Bit | 0 bis 7 | Status | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das DIR-Status-Flag (Zählrichtung) wird durch das HSC-Subsystem gesteuert. Bei Aufwärtszählung des HSC-Istwerts wird das Richtungs- Flag gesetzt (1). Bei Abwärtszählung des HSC-Istwerts wird das Richtungs-Flag gelöscht (0).

Wenn die Istwert-Zählung angehalten wird, behält das Richtungs-Bit seinen Wert bei. Das Richtungs-Flag wird nur geändert, wenn die Richtung der Istwert-Zählung umgekehrt wird.

Dieses Bit wird durch das HSC-Subsystem kontinuierlich aktualisiert, wenn sich die Steuerung in einem Ausführungsmodus befindet.

## Modus fertig (MD)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| MD - Modus fertig | HSC:0/MD | Bit | 0 oder 1 | Status | Lesen/Schreiben |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das MD-Status-Flag (Modus fertig) wird durch das HSC-Subsystem gesetzt (1), wenn der HSC-Modus 0 oder 1 aktiviert wurde und der Istwert bis zum oberen Sollwert aufwärts gezählt wird.

## Abwärtszähler (CD)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| CD - Abwärtszähler | HSC:0/CD | Bit | 2 bis 7 | Status | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)" auf Seite 5-16.

Das CD-Bit (Abwärtszähler) wird bei bidirektionalen Zählern (Modi 2 bis 7) verwendet. Wenn das CE-Bit gesetzt ist, wird auch das CD-Bit gesetzt (1). Wenn das CE-Bit gelöscht ist, wird auch das CD-Bit gelöscht (0).

## Aufwärtszähler (CU)

| Beschreibung | Adresse | Datenformat | HSC-Modi(1) | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| CU - Aufwärtszähler | HSC:0/CU | Bit | 0 bis 7 | Status | Nur Lesen |

(1) Beschreibungen der Modi finden Sie im Abschnitt „HSC-Modus (MOD)“ auf Seite 5-16.

Das CU-Bit (Aufwärtszähler) wird bei allen HSC (Modi 0 bis 7) verwendet. Wenn das CE-Bit gesetzt ist, wird auch das CU-Bit gesetzt (1). Wenn das CE-Bit gelöscht ist, wird auch das CU-Bit gelöscht (0).

## HSC-Modus (MOD)

| Beschreibung | Adresse | Datenformat | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| MOD - HSC-Modus | HSC:0.MOD | Wort (INT) | Steuerung | Nur Lesen |

Die MOD-Variable (Modus) legt eine der acht Betriebsarten für den Hochgeschwindigkeitszähler fest. Dieser ganzzahlige Wert wird über das Programmiergerät konfiguriert; im Steuerprogramm ist diese Variable schreibgeschützt.

**Tabelle 5.3 HSC-Betriebsmodi**

| Modusnummer | Typ |
|---|---|
| 0 | Aufwährtszähler - Der Istwert wird bei Erreichen des oberen Sollwerts sofort gelöscht (0). In diesem Modus kann kein unterer Sollwert definiert werden. |
| 1 | Aufwährtszähler mit externem Rücksetzen und Halten - Der Istwert wird bei Erreichen des oberen Sollwerts sofort gelöscht (0). In diesem Modus kann kein unterer Sollwert definiert werden. |
| 2 | Zähler mit externer Richtung |
| 3 | Zähler mit externer Richtung, Rücksetzen und Halten |
| 4 | Zwei Eingangszähler (aufwärts und abwärts) |
| 5 | Zwei Eingangszähler (aufwärts und abwärts) mit externem Rücksetzen und Halten |
| 6 | 2-Kanal-Zähler (Eingänge A und B phasenverschoben) |
| 7 | 2-Kanal-Zähler (Eingänge A und B phasenverschoben) mit externem Rücksetzen und Halten |

*HSC-Modus 0 - Aufwärtszähler*

**Tabelle 5.4 HSC-Modus 0, Beispiele(1)**

| **Eingangs-klemmen** | I1:0.0/0 (HSC0)<br>I1:0.0/4 (HSC1) | | | | I1:0.0/1 (HSC0)<br>I1:0.0/5 (HSC1) | | | | I1:0.0/2 (HSC0)<br>I1:0.0/6 (HSC1) | | | | I1:0.0/3 (HSC0)<br>I1:0.0/7 (HSC1) | | | | **CE-Bit** | **Anmerkungen** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Funktion** | **Zählwert** | | | | **nicht verwendet** | | | | **nicht verwendet** | | | | **nicht verwendet** | | | | | |
| **Beispiel 1** | ⇑ | | | | | | | | | | | | | | | | ein (1) | HSC-Istwert + 1 Zählung |
| **Beispiel 2** | ⇑ | ein (1) | ⇓ | aus (0) | | | | | | | | | | | | | aus (0) | Istwert halten |

(1) HSC1 nur bei MicroLogix 1500.

Leere Zellen = nicht von Belang, ⇑ = Anstiegsflanke, ⇓ = abfallende Flanke

**HINWEIS** Eingänge I1:0.0/0 bis I1:0.0/7 sind unabhängig von der Verwendung eines Hochgeschwindigkeitszählers als Eingänge für andere Funktionen verfügbar.

*HSC-Modus 1 - Aufwärtszähler mit externem Rücksetzen und Halten*

**Tabelle 5.5 HSC-Modus 1, Beispiele(1)**

| **Eingangs-klemmen** | I1:0.0/0 (HSC0)<br>I1:0.0/4 (HSC1) | | | | I1:0.0/1 (HSC0)<br>I1:0.0/5 (HSC1) | | | | I1:0.0/2 (HSC0)<br>I1:0.0/6 (HSC1) | | | | I1:0.0/3 (HSC0)<br>I1:0.0/7 (HSC1) | | | | **CE-Bit** | **Anmerkungen** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Funktion** | **Zählwert** | | | | **nicht verwendet** | | | | **Reset** | | | | **Halten** | | | | | |
| **Beispiel 1** | ⇑ | | | | | | | | | ein (1) | ⇓ | aus (0) | | | | aus (0) | ein (1) | HSC-Istwert + 1 Zählung |
| **Beispiel 2** | | | | | | | | | | ein (1) | ⇓ | aus (0) | | ein (1) | | | | Istwert halten |
| **Beispiel 3** | | | | | | | | | | ein (1) | ⇓ | aus (0) | | | | | aus (0) | Istwert halten |
| **Beispiel 4** | | ein (1) | ⇓ | aus (0) | | | | | | ein (1) | ⇓ | aus (0) | | | | | | Istwert halten |
| **Beispiel 5** | | | | | | | | | ⇑ | | | | | | | | | Istwert löschen (=0) |

(1) HSC1 nur bei MicroLogix 1500.

Leere Zellen = nicht von Belang, ⇑ = Anstiegsflanke, ⇓ = abfallende Flanke

**HINWEIS** Eingänge I1:0.0/0 bis I1:0.0/7 sind unabhängig von der Verwendung eines Hochgeschwindigkeitszählers als Eingänge für andere Funktionen verfügbar.

*HSC-Modus 2 - Zähler mit externer Richtung*

**Tabelle 5.6 HSC-Modus 2, Beispiele(1)**

| **Eingangs-klemmen** | I1:0.0/0 (HSC0)<br>I1:0.0/4 (HSC1) | | | | I1:0.0/1 (HSC0)<br>I1:0.0/5 (HSC1) | | | | I1:0.0/2 (HSC0)<br>I1:0.0/6 (HSC1) | | | | I1:0.0/3 (HSC0)<br>I1:0.0/7 (HSC1) | | | | **CE-Bit** | **Anmerkungen** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Funktion** | **Zählwert** | | | | **Richtung** | | | | **nicht verwendet** | | | | **nicht verwendet** | | | | | |
| **Beispiel 1** | ⇑ | | | | | | | aus (0) | | | | | | | | | ein (1) | HSC-Istwert + 1 Zählung |
| **Beispiel 2** | ⇑ | | | | | ein (1) | | | | | | | | | | | ein (1) | HSC-Istwert - 1 Zählung |
| **Beispiel 3** | | | | | | | | | | | | | | | | | aus (0) | Istwert halten |

(1) HSC1 nur bei MicroLogix 1500.

Leere Zellen = nicht von Belang, ⇑ = Anstiegsflanke, ⇓ = abfallende Flanke

**HINWEIS** Eingänge I1:0.0/0 bis I1:0.0/7 sind unabhängig von der Verwendung eines Hochgeschwindigkeitszählers als Eingänge für andere Funktionen verfügbar.

*HSC-Modus 3 - Zähler mit externer Richtung, Rücksetzen und Halten*

**Tabelle 5.7 HSC-Modus 3, Beispiele(1)**

| **Eingangs-klemmen** | I1:0.0/0 (HSC0)<br>I1:0.0/4 (HSC1) | | | | I1:0.0/1 (HSC0)<br>I1:0.0/5 (HSC1) | | | | I1:0.0/2 (HSC0)<br>I1:0.0/6 (HSC1) | | | | I1:0.0/3 (HSC0)<br>I1:0.0/7 (HSC1) | | | | **CE-Bit** | **Anmerkungen** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Funktion** | **Zählwert** | | | | **Richtung** | | | | **Reset** | | | | **Halten** | | | | | |
| **Beispiel 1** | ⇑ | | | | | | | aus (0) | | ein (1) | ⇓ | aus (0) | | | | aus (0) | ein (1) | HSC-Istwert + 1 Zählung |
| **Beispiel 2** | ⇑ | | | | | ein (1) | | | | ein (1) | ⇓ | aus (0) | | | | aus (0) | ein (1) | HSC-Istwert - 1 Zählung |
| **Beispiel 3** | | | | | | | | | | ein (1) | ⇓ | aus (0) | | ein (1) | | | | Istwert halten |
| **Beispiel 4** | | | | | | | | | | ein (1) | ⇓ | aus (0) | | | | | aus (0) | Istwert halten |
| **Beispiel 5** | | ein (1) | ⇓ | aus (0) | | | | | | ein (1) | ⇓ | aus (0) | | | | | | Istwert halten |
| **Beispiel 6** | | | | | | | | | ⇑ | | | | | | | | | Istwert löschen (=0) |

(1) HSC1 nur bei MicroLogix 1500.

Leere Zellen = nicht von Belang, ⇑ = Anstiegsflanke, ⇓ = abfallende Flanke

**HINWEIS** Eingänge I1:0.0/0 bis I1:0.0/7 sind unabhängig von der Verwendung eines Hochgeschwindigkeitszählers als Eingänge für andere Funktionen verfügbar.

*HSC-Modus 4 - Zwei Eingangszähler (aufwärts und abwärts)*

**Tabelle 5.8 HSC-Modus 4, Beispiele(1)**

| **Eingangsklemmen** | I1:0.0/0 (HSC0)<br>I1:0.0/4 (HSC1) | | | | I1:0.0/1 (HSC0)<br>I1:0.0/5 (HSC1) | | | | I1:0.0/2 (HSC0)<br>I1:0.0/6 (HSC1) | | | | I1:0.0/3 (HSC0)<br>I1:0.0/7 (HSC1) | | | | **CE-Bit** | **Anmerkungen** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Funktion** | **Aufwärts-zählung** | | | | **Abwärts-zählung** | | | | **nicht verwendet** | | | | **nicht verwendet** | | | | | |
| **Beispiel 1** | ⇑ | | | | | ein (1) | ⇓ | aus (0) | | | | | | | | | ein (1) | HSC-Istwert + 1 Zählung |
| **Beispiel 2** | | ein (1) | ⇓ | aus (0) | ⇑ | | | | | | | | | | | | ein (1) | HSC-Istwert - 1 Zählung |
| **Beispiel 3** | | | | | | | | | | | | | | | | | aus (0) | Istwert halten |

(1) HSC1 nur bei MicroLogix 1500.

Leere Zellen = nicht von Belang, ⇑ = Anstiegsflanke, ⇓ = abfallende Flanke

**HINWEIS** Eingänge I1:0.0/0 bis I1:0.0/7 sind unabhängig von der Verwendung eines Hochgeschwindigkeitszählers als Eingänge für andere Funktionen verfügbar.

*HSC-Modus 5 - Zwei Eingangszähler (aufwärts und abwärts) mit externem Rücksetzen und Halten*

**Tabelle 5.9 HSC-Modus 5, Beispiele(1)**

| **Eingangs-klemmen** | I1:0.0/0 (HSC0)<br>I1:0.0/4 (HSC1) | | | | I1:0.0/1 (HSC0)<br>I1:0.0/5 (HSC1) | | | | I1:0.0/2 (HSC0)<br>I1:0.0/6 (HSC1) | | | | I1:0.0/3 (HSC0)<br>I1:0.0/7 (HSC1) | | | | **CE-Bit** | **Anmerkungen** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Funktion** | **Zählwert** | | | | **Richtung** | | | | **Reset** | | | | **Halten** | | | | | |
| **Beispiel 1** | ⇑ | | | | | ein (1) | ⇓ | aus (0) | | ein (1) | ⇓ | aus (0) | | | | aus (0) | ein (1) | HSC-Istwert + 1 Zählung |
| **Beispiel 2** | | ein (1) | ⇓ | aus (0) | ⇑ | | | | | ein (1) | ⇓ | aus (0) | | | | aus (0) | ein (1) | HSC-Istwert - 1 Zählung |
| **Beispiel 3** | | | | | | | | | | ein (1) | ⇓ | aus (0) | | ein (1) | | | | Istwert halten |
| **Beispiel 4** | | | | | | | | | | ein (1) | ⇓ | aus (0) | | | | | aus (0) | Istwert halten |
| **Beispiel 5** | | ein (1) | ⇓ | aus (0) | | | | | | ein (1) | ⇓ | aus (0) | | | | | | Istwert halten |
| **Beispiel 6** | | | | | | | | | ⇑ | | | | | | | | | Istwert löschen (=0) |

(1) HSC1 nur bei MicroLogix 1500.

Leere Zellen = nicht von Belang, ⇑ = Anstiegsflanke, ⇓ = abfallende Flanke

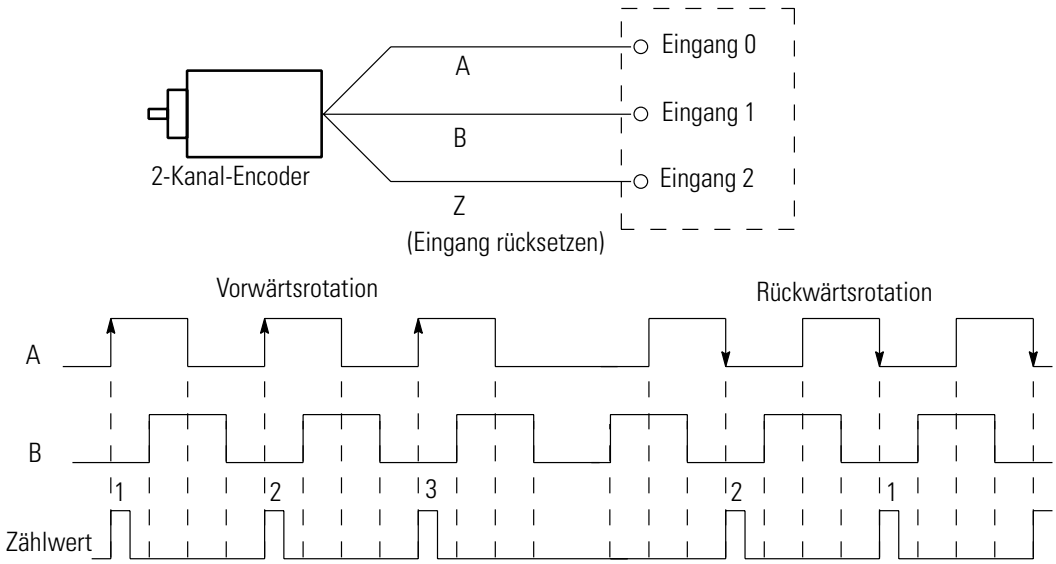
**HINWEIS** Eingänge I1:0.0/0 bis I1:0.0/7 sind unabhängig von der Verwendung eines Hochgeschwindigkeitszählers als Eingänge für andere Funktionen verfügbar.

### *2-Kanal-Encoder verwenden*

Der 2-Kanal-Encoder wird verwendet, um die Drehrichtung und Drehposition bei Maschinen wie einer Drehbank zu ermitteln. Der bidirektionale Zähler zählt die Umdrehungen des 2-Kanal-Encoders.

Die nachfolgende Abbildung zeigt einen 2-Kanal-Encoder, der an die Eingänge 0, 1 und 2 angeschlossen ist. Die Zählrichtung wird über den Phasenwinkel zwischen A und B ermittelt. Wenn A vor B liegt, wird aufwärts gezählt. Wenn B vor A liegt, wird abwärts gezählt.

Der Zähler kann über Eingang Z rückgesetzt werden. Die Z-Ausgänge der Encoder liefern normalerweise einen Impuls pro Umdrehung.




*HSC-Modus 6 – 2-Kanal-Zähler (Eingänge A und B phasenverschoben)*

**Tabelle 5.10 HSC-Modus 6, Beispiele(1)**

| **Eingangs-klemmen** | I1:0.0/0 (HSC0)<br>I1:0.0/4 (HSC1) | | | | I1:0.0/1 (HSC0)<br>I1:0.0/5 (HSC1) | | | | I1:0.0/2 (HSC0)<br>I1:0.0/6 (HSC1) | | | | I1:0.0/3 (HSC0)<br>I1:0.0/7 (HSC1) | | | | **CE-Bit** | **Anmerkungen** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Funktion** | **Zählwert A** | | | | **Zählwert B** | | | | **nicht verwendet** | | | | **nicht verwendet** | | | | | |
| **Beispiel 1(2)** | ⇑ | | | | | | | aus (0) | | | | | | | | | ein (1) | HSC-Istwert + 1 Zählung |
| **Beispiel 2(3)** | | | ⇓ | | | | | aus (0) | | | | | | | | | ein (1) | HSC-Istwert - 1 Zählung |
| **Beispiel 3** | | | | aus (0) | | | | | | | | | | | | | | Istwert halten |
| **Beispiel 4** | | ein (1) | | | | | | | | | | | | | | | | Istwert halten |
| **Beispiel 5** | | | | | | ein (1) | | | | | | | | | | | | Istwert halten |
| **Beispiel 6** | | | | | | | | | | | | | | | | | aus (0) | Istwert halten |

(1) HSC1 nur bei MicroLogix 1500.

(2) Zähleingang A liegt vor Zähleingang B.

(3) Zähleingang B liegt vor Zähleingang A.

Leere Zellen = nicht von Belang, ⇑ = Anstiegsflanke, ⇓ = abfallende Flanke

**HINWEIS** Eingänge I1:0.0/0 bis I1:0.0/7 sind unabhängig von der Verwendung eines Hochgeschwindigkeitszählers als Eingänge für andere Funktionen verfügbar.

*HSC-Modus 7 - 2-Kanal-Zähler (Eingänge A und B phasenverschoben) mit externem Rücksetzen und Halten*

**Tabelle 5.11 HSC-Modus 7, Beispiele(1)**

| **Eingangs-klemmen** | I1:0.0/0 (HSC0)<br>I1:0.0/4 (HSC1) | | | | I1:0.0/1 (HSC0)<br>I1:0.0/5 (HSC1) | | | | I1:0.0/2 (HSC0)<br>I1:0.0/6 (HSC1) | | | | I1:0.0/3 (HSC0)<br>I1:0.0/7 (HSC1) | | | | **CE-Bit** | **Anmerkungen** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Funktion** | **Zählwert A** | | | | **Zählwert B** | | | | **Z Rücksetzen** | | | | **Halten** | | | | | |
| **Beispiel 1(2)** | ⇑ | | | | | | | aus (0) | | | | | | | | aus (0) | ein (1) | HSC-Istwert + 1 Zählung |
| **Beispiel 2(3)** | | | ⇓ | | | | | aus (0) | | | | aus (0) | | | | aus (0) | ein (1) | HSC-Istwert - 1 Zählung |
| **Beispiel 3** | | | ⇓ | aus (0) | | | | aus (0) | | ein (1) | | | | | | | | Istwert auf null rücksetzen |
| **Beispiel 4** | | ein (1) | | | | | | | | | | | | | | | | Istwert halten |
| **Beispiel 5** | | | | | | ein (1) | | | | | | | | | | | | Istwert halten |
| **Beispiel 6** | | | | | | | | | | | | aus (0) | | ein (1) | | | | Istwert halten |
| **Beispiel 7** | | | | | | | | | | | | aus (0) | | | | | aus (0) | Istwert halten |

(1) HSC1 nur bei MicroLogix 1500.

(2) Zähleingang A liegt vor Zähleingang B.

(3) Zähleingang B liegt vor Zähleingang A.

Leere Zellen = nicht von Belang, ⇑ = Anstiegsflanke, ⇓ = abfallende Flanke

**HINWEIS** Eingänge I1:0.0/0 bis I1:0.0/7 sind unabhängig von der Verwendung eines Hochgeschwindigkeitszählers als Eingänge für andere Funktionen verfügbar.

## Istwert (ACC)

| Beschreibung | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| ACC - Istwert | HSC:0.ACC | Doppelwort (32-Bit INT) | Steuerung | Lesen/Schreiben |

Der ACC (Istwert) enthält die Anzahl der Zählschritte, die von dem HSC-Subsystem erkannt wurden. Wenn der Modus 0 oder 1 kon-figuriert ist, wird der Software-Istwert gelöscht (0), wenn ein oberer Sollwert erreicht oder eine Überlaufbedingung festgestellt wird.

## Oberer Sollwert (HIP)

| Beschreibung | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| HIP - Oberer Sollwert | HSC:0.HIP | Doppelwort (32-Bit INT) | Steuerung | Lesen/Schreiben |

Über den HIP (oberen Sollwert) wird festgelegt, wann das HSC-Sub- system einen Interrupt erzeugt. Zum Laden von Daten in den oberen Sollwert stehen verschiedene Möglichkeiten im Steuerprogramm zur Verfügung:

- Steuer-Bit „Parameter einstellen“ (HSC:0/SP) umschalten (tief nach hoch). Wenn das SP-Bit auf hoch gesetzt wird, werden die aktuell in dem HSC-Funktionsfile gespeicherten Daten in das HSC-Sub- system übertragen/geladen.
- Neue HSC-Parameter mit dem HSL-Befehl laden. Siehe „HSL – Hochgeschwindigkeitszähler laden“ auf Seite 5-26.

Die Daten, die in den oberen Sollwert geladen werden, müssen kleiner als oder gleich dem Wert des Überlaufparameters (HSC:0.OVF) sein; andernfalls wird ein HSC-Fehler generiert.

## Unterer Sollwert (LOP)

| Beschreibung | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| LOP - Unterer Sollwert | HSC:0.LOP | Doppelwort (32-Bit INT) | Steuerung | Lesen/Schreiben |

Über den LOP (unteren Sollwert) wird festgelegt, wann das HSC-Sub- system einen Interrupt erzeugt. Zum Laden von Daten in den unteren Sollwert stehen verschiedene Möglichkeiten im Steuerprogramm zur Verfügung:

- Steuer-Bit „Parameter einstellen“ (HSC:0/SP) umschalten (tief nach hoch). Wenn das SP-Bit auf hoch gesetzt wird, werden die aktuell in dem HSC-Funktionsfile gespeicherten Daten in das HSC-Sub- system übertragen/geladen.
- Neue HSC-Parameter mit dem HSL-Befehl laden. Siehe „HSL – Hochgeschwindigkeitszähler laden“ auf Seite 5-26.

Die Daten, die in den unteren Sollwert geladen werden, müssen größer als oder gleich dem Wert des Unterlaufparameters (HSC:0.UNF) sein; andernfalls

wird ein HSC-Fehler generiert. (Wenn der Unter-laufwert und der untere Sollwert negativ sind, muss der untere Sollwert eine Zahl mit einem kleineren absoluten Wert sein.)

## Überlauf (OVF)

| Beschreibung | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| OVF - Überlauf | HSC:0.OVF | Doppelwort (32-Bit INT) | Steuerung | Lesen/Schreiben |

Der OVF (Überlauf) legt die obere Zählgrenze für den Zähler fest. Wenn der Zähler-Istwert den in dieser Variablen angegebenen Wert überschreitet, wird ein Überlauf-Interrupt generiert. Das HSC-Sub- system setzt den Istwert auf den Unterlaufwert, und der Zähler setzt die Zählung ausgehend von dem Unterlaufwert fort (bei diesem Übergang gehen keine Zählschritte verloren). Als Überlaufwert kann jeder beliebige Wert angegeben werden, der größer als der Unterlauf- wert ist und zwischen -2.147.483.648 und 2.147.483.647 liegt.

Zum Laden von Daten in die Überlaufvariable muss das Steuer-Bit Parameter einstellen (HSC:0.0/SP) durch das Steuerprogramm umgeschaltet werden (tief nach hoch). Wenn das SP-Bit auf hoch gesetzt wird, werden die aktuell in dem HSC-Funktionsfile gespeicherten Daten in das HSC-Subsystem übertragen/ geladen.

**HINWEIS** Die Daten, die in die Überlaufvariable geladen werden, müssen größer als der Wert des oberen Sollwerts (HSC:0.HIP) sein; andernfalls wird ein HSC-Fehler generiert.

## Unterlauf (UNF)

| Beschreibung | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| UNF - Unterlauf | HSC:0.UNF | Doppelwort (32-Bit INT) | Steuerung | Lesen/Schreiben |

Der UNF (Unterlauf) legt die untere Zählgrenze für den Zähler fest. Wenn der Zähler-Istwert den in dieser Variablen angegebenen Wert unterschreitet, wird ein Unterlauf-Interrupt generiert. Dann setzt das HSC-Subsystem den Istwert auf den Überlaufwert, und der Zähler setzt die Zählung ausgehend von dem Überlaufwert fort (bei diesem Übergang gehen keine Zählschritte verloren). Als Unterlaufwert kann jeder beliebige Wert angegeben werden, der kleiner als der Überlauf- wert ist und zwischen -2.147.483.648 und 2.147.483.647 liegt.

Zum Laden von Daten in die Unterlaufvariable muss das Steuer-Bit „Parameter einstellen“ (HSC:0.0/SP) durch das Steuerprogramm umgeschaltet werden (tief nach hoch). Wenn das SP-Bit auf hoch gesetzt wird, werden die aktuell in dem HSC-Funktionsfile gespeicherten Daten in das HSC-Subsystem übertragen/geladen.

**HINWEIS** Die Daten, die in die Überlaufvariable geladen werden, müssen größer als der Wert des oberen Sollwerts (HSC:0.HIP) sein; andernfalls wird ein HSC-Fehler generiert.

## Ausgangsmaske (OMB)

| Beschreibung | Adresse | Datenformat | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| OMB - Ausgangsmaske | HSC:0.OMB | Wort (16-Bit-Binärwert) | Steuerung | Nur Lesen |

Über die OMB (Ausgangsmaske) werden die Ausgänge auf der Steuerung festgelegt, die direkt durch den Hochgeschwindigkeits- zähler gesteuert werden. Das HSC-Subsystem kann Ausgänge direkt (ohne Eingriff des Steuerprogramms) ein- oder ausschalten, sobald der HSC-Istwert den oberen oder unteren Sollwert erreicht. Durch das Bitmuster, das in der OMB-Variable gespeichert ist, werden die Ausgänge festgelegt, die durch den HSC gesteuert werden.

Das Bitmuster der OMB-Variable entspricht direkt den Ausgangs-Bits der Steuerung. Bits, die gesetzt wurden (1), sind aktiviert und können durch das HSC-Subsystem ein- und ausgeschaltet werden. Bits, die nicht gesetzt wurden (0), können nicht durch das HSC-Subsystem ein- und ausgeschaltet werden. Das Bitmuster der Maske kann nur bei der erstmaligen Konfiguration festgelegt werden.

Die nachfolgende Tabelle zeigt diesen Zusammenhang:

**Tabelle 5.12 Auswirkung der HSC-Ausgangsmaske auf Ausgänge der Basiseinheit**

| Ausgangsadresse | 16-Bit-Datenwort mit ganzzahligem Vorzeichen | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| HSC:0.HPO (Ausgang bei oberem Sollwert) | | | | | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| | | | | | | | | | | | | | | | | |
| HSC:0.OMB (Ausgangsmaske) | | | | | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| | | | | | | | | | | | | | | | | |
| O0:0.0 | | | | | **0** | | | | | **0** | **0** | **1** | | | **0** | **1** |
| | | | | | | | | | | | | | | | | |

Die Ausgänge in den schwarz markierten Feldern werden durch das HSC-Subsystem gesteuert. In der Maske wird festgelegt, welche Ausgänge gesteuert werden können. Der Ausgang bei oberem Sollwert oder der Ausgang bei unterem Sollwert (HPO oder LPO) legen fest, ob die einzelnen Ausgänge ein- (1) oder ausgeschaltet (0) sind. Anders ausgedrückt: Der Ausgang bei oberem oder unterem Sollwert wird durch die Ausgangsmaske geführt, die als Filter fungiert.

Die Bits in den grau markierten Feldern werden nicht benutzt. Die ersten 12 Bits des Maskenworts werden verwendet; die übrigen Masken-Bits können nicht verwendet werden, da sie keinem physischen Ausgang auf der Basiseinheit zugeordnet sind.

Das Bitmuster der Maske kann nur bei der erstmaligen Konfiguration festgelegt werden.

## Ausgang bei oberem Sollwert (HPO)

| Beschreibung | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| HPO - Ausgang bei oberem Sollwert | HSC:0.HPO | Wort (16-Bit-Binärwert) | Steuerung | Lesen/Schreiben |

Der HPO (Ausgang bei oberem Sollwert) definiert den Status (1 = EIN oder 0 = AUS) der Ausgänge auf der Steuerung, wenn der obere Sollwert erreicht wird. Weitere Informationen dazu, wie Ausgänge basierend darauf direkt ein- oder ausgeschaltet werden können, ob die obere Voreinstellung erreicht wurde, finden Sie im Abschnitt „Ausgangsmaske (OMB)“ auf Seite 5-24.

Das Bitmuster für den Ausgang bei oberem Sollwert kann während der erstmaligen Konfiguration oder während des Betriebs der Steuerung festgelegt werden. Verwenden Sie den HSL-Befehl oder das SP-Bit, um während des Betriebs der Steuerung neue Parameter zu laden.

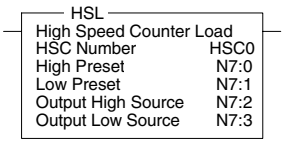
## Ausgang bei unterem Sollwert (LPO)

| Beschreibung | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| LPO - Ausgang bei unterem Sollwert | HSC:0.LPO | Wort (16-Bit-Binärwert) | Steuerung | Lesen/Schreiben |

Der LPO (Ausgang bei unterem Sollwert) definiert den Status (1 = EIN oder 0 = AUS) der Ausgänge auf der Steuerung, wenn der untere Sollwert erreicht wird. Weitere Informationen dazu, wie Ausgänge basierend darauf direkt ein- oder ausgeschaltet werden können, ob die untere Voreinstellung erreicht wurde, finden Sie im Abschnitt „Ausgangsmaske (OMB)“ auf Seite 5-24.

Das Bitmuster für den Ausgang bei unterem Sollwert kann während der erstmaligen Konfiguration oder während des Betriebs der Steuerung festgelegt werden. Verwenden Sie den HSL-Befehl oder das SP-Bit, um während des Betriebs der Steuerung neue Parameter zu laden.

# HSL – Hochgeschwindigkeitszähler laden




Befehlstyp: Ausgang

| Steuerung | Datengröße | Ausführungszeit bei Strompfad: | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 46,7 µs | 0,0 µs |
| | Doppelwort | 47,3 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 39,7 µs | 0,0 µs |
| | Doppelwort | 40,3 µs | 0,0 µs |

Mit dem HSL-Befehl (Hochgeschwindigkeitszähler laden) werden die oberen und unteren Sollwerte und die oberen und unteren Ausgangs-Quellen auf einen Hochgeschwindigkeitszähler angewendet. Diese Parameter werden nachfolgend beschrieben:

- Zählernummer – Gibt an, welcher Hochgeschwindigkeitszähler verwendet wird. 0 = HSC0 und 1 = HSC1 (*nur MicroLogix 1500*).
- Oberer Sollwert - Gibt den Wert im oberen Sollwert-Register an. Die Werte für den oberen Sollwert liegen in dem Bereich von -32786 bis 32767 (Wort) und -2.147.483.648 bis 2.147.483.647 (Doppelwort).
- Unterer Sollwert - Gibt den Wert im unteren Sollwert-Register an. Die Werte für den unteren Sollwert liegen in dem Bereich von -32786 bis 32767 (Wort) und -2.147.483.648 bis 2.147.483.647 (Doppelwort).
- Ausgang obere Quelle – Gibt den Wert im Ausgangsregister des oberen Sollwerts (HPO) an. Die Werte für die obere Ausgangs-quelle liegen in einem Bereich von 0 bis 65.535.
- Ausgang untere Quelle – Gibt den Wert im Ausgangsregister des unteren Sollwerts (LPO) an. Die Werte für die untere Ausgangs-quelle liegen in einem Bereich von 0 bis 65.535.

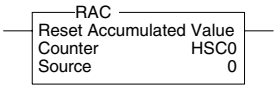
Die gültigen Adressierungsmodi und Filetypen werden in der nachfolgenden Tabelle dargestellt:

**Tabelle 5.13 Gültige Adressierungsmodi und Filetypen für den HSL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS – Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Zählernummer | | | | | | | | | | | | | | | | | | | | | | | | • | | | | | | |
| Oberer Sollwert | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Unterer Sollwert | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Obere Ausgangsquelle | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Untere Ausgangsquelle | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |

# RAC – Istwert zurücksetzen




Befehlstyp: Ausgang

| Steuerung | Ausführungszeit bei Strompfad: | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 21,2 µs | 0,0 µs |
| MicroLogix 1500 | 17,8 µs | 0,0 µs |

Mit dem RAC-Befehl wird der Hochgeschwindigkeitszähler zurückgesetzt und ein bestimmter Wert in den HSC-Istwert geschrieben. Der RAC-Befehl wird mit folgenden Parametern verwendet:

- Zählernummer - Gibt an, welcher Hochgeschwindigkeitszähler verwendet wird.
  - Zählernummer 0 = HSC0 (*nur MicroLogix 1200 und 1500*)
  - Zählernummer 1 = HSC1 (*nur MicroLogix 1500*)
- Quelle - Gibt den Standort der Daten an, die in den HSC-Istwert geladen werden sollen. Die Daten liegen in einem Bereich von -2.147.483.648 bis 2.147.483.647.

Die gültigen Adressierungsmodi und Filetypen werden in der nachfolgenden Tabelle dargestellt:

**Tabelle 5.14 Gültige Adressierungsmodi und Filetypen für den RAC-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CSF - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Zählernummer | | | | | | | | | | | | | | | | | | | | | | | | • | | | | | | |
| Quelle | | | | | | | | | • | | | | | | | | | | | | | | | • | • | • | | | • | |

# Programmierbarer Endschalter-File (PLS)

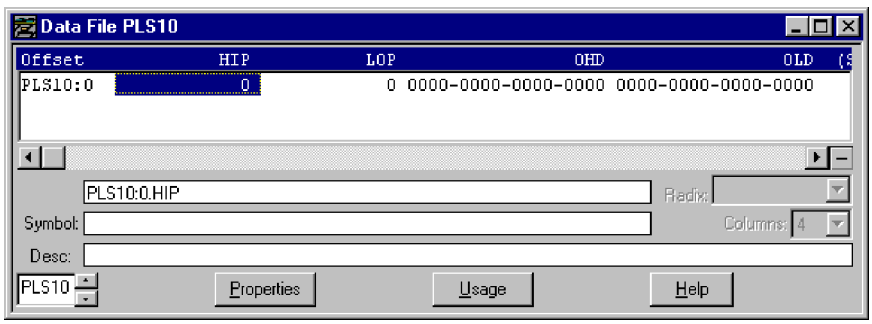
Mit der Funktion für programmierbare Endschalter können Sie den Hochgeschwindigkeitszähler so konfigurieren, dass er als program-mierbarer Endschalter (PLS) oder als Nockendrehschalter arbeitet.

Wenn der PLS-Betrieb aktiviert ist, verwendet der Hochgeschwindig-keitszähler (HSC) ein PLS-Datenfile für End-/Nockenpositionen. Jede End-/Nockenposition verfügt über entsprechende Datenparameter, die dazu verwendet werden, die physischen Ausgänge an der Basiseinheit der Steuerung zurückzusetzen oder zu setzen. In der nachfolgenden Abbildungen ist ein PLS-Datenfile dargestellt.

**WICHTIG** Die PLS-Funktion arbeitet nur zusammen mit dem HSC einer MicroLogix 1200- oder 1500-Steuerung. Bevor Sie die PLS-Funktion verwenden können, muss der HSC konfiguriert werden.

## PLS-Datenfile

Datenfiles 9 bis 255 können für den PLS-Betrieb verwendet werden. Jeder PLS-Datenfile kann aus maximal 256 Elementen bestehen. Jedes Element eines PLS-Files benötigt 6 Anwenderwörter an Speicher. Nachfolgend wird eine Abbildung eines PLS-Datenfiles gezeigt:




## PLS-Betrieb

Wenn die PLS-Funktion deaktiviert ist und die Steuerung sich im Run-Modus befindet, zählt der HSC die eingehenden Impulse. Sobald die Zählung den ersten Sollwert erreicht hat (HIP – oberer Eingangs- Sollwert oder LOP – unterer Eingangs-Sollwert), der im PLS-File festgelegt ist, werden die Ausgangsquelldaten (OHD – obere Aus- gangsdaten oder OLD – untere Ausgangsdaten) durch die HSC-Maske geschrieben.

Zu diesem Zeitpunkt wird der nächste Sollwert (HIP oder LOP) aktiviert, der in dem PLS-File festgelegt wurde.

Wenn der HSC bis zu diesem neuen Sollwert zählt, werden die neuen Ausgangsdaten durch die HSC-Maske geschrieben. Dieser Vorgang wiederholt sich so lange, bis das letzte Element in dem PLS-File geladen wurde. Zu diesem Zeitpunkt wird das aktive Element in dem PLS-File auf Null zurückgesetzt. Dieses Verhalten stellt einen kontinuierlichen Betrieb dar.

**HINWEIS** Die oberen Ausgangsdaten (OHD) werden nur dann geschrieben, wenn der obere Eingangs-Sollwert (HIP) erreicht wird. Die unteren Ausgangsdaten (OLD) werden nur dann geschrieben, wenn der untere Sollwert erreicht wird.

**HINWEIS** Die oberen Ausgangsdaten sind nur dann betriebs-fähig, wenn der Zähler aufwärts zählt. Die unteren Ausgangsdaten sind nur dann betriebsfähig, wenn der Zähler abwärts zählt.

Wenn während des Betriebs ungültige Daten geladen werden, tritt ein HSC-Fehler innerhalb des HSC-Funktionsfiles auf. Dieser Fehler be-wirkt jedoch keinen Steuerungsausfall. Wenn ein ungültiger Parameter entdeckt wurde, wird er übersprungen und der nächste Parameter wird zur Ausführung geladen (sofern er gültig ist).

Sie können den PLS nach oben oder nach unten als auch in beiden Richtungen verwenden. Wenn Ihre Anwendung nur in eine Richtung zählt, ignorieren Sie die anderen Parameter.

Die PLS-Funktion kann nur mit allen anderen HSC-Funktionen in Betrieb genommen werden. Die Möglichkeiten zum Auswählen, welche HSC-Ereignisse einen Benutzer-Interrupt hervorrufen sollen, sind unbegrenzt.

## Adressierung von PLS-Files

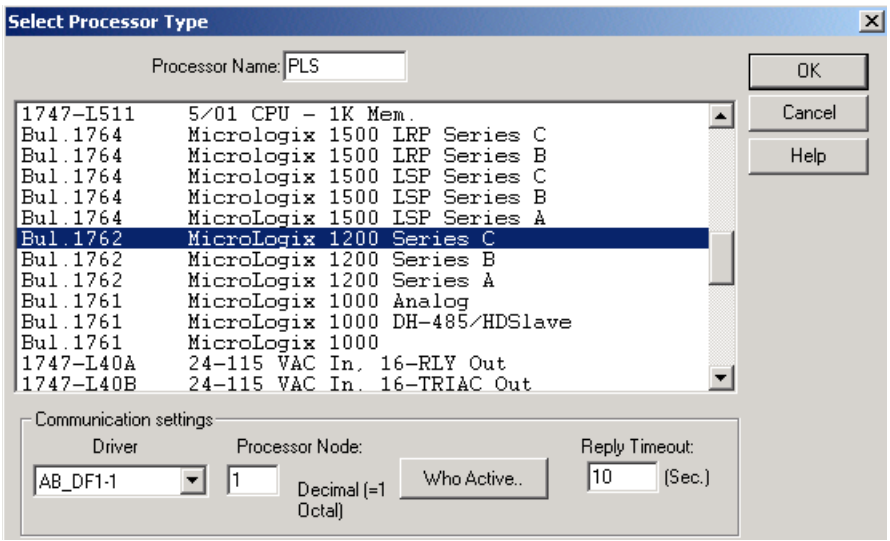
Im Nachfolgenden wird das Adressierformat für einen PLS-File dargestellt.

| **Format** | **Bedeutung** | | |
|---|---|---|---|
| **PLSf:e.s** | **PLS** | Programmierbarer Endschalter-File | |
| | **F** | Filenummer | Die Filenummer kann aus einem Bereich von 9 bis 255 ausgewählt werden. |
| | **:** | Elementendezeichen | |
| | **e** | Elementnummer | Die Elementnummer kann aus einem Bereich von 0 bis 255 ausgewählt werden. |
| | **.** | Unterelement-Endezeichen | |
| | **S** | Unterelementnummer | Der gültige Wertbereich für Unterelementnummern ist 0 bis 5 |
| **Beispiele:** | **PLS10:2**<br>**PLS12:36.5** | | **PLS-File 10, Element 2**<br>**PLS-File 12, Element 36, Unterelement 5 (Ausgang untere Quelle)** |

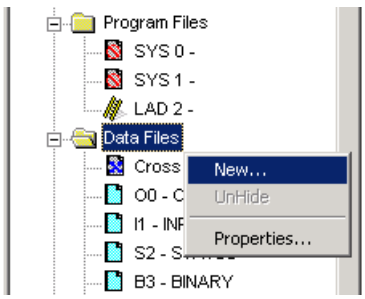
## Beispiel für PLS

*Einrichten des PLS-Files*

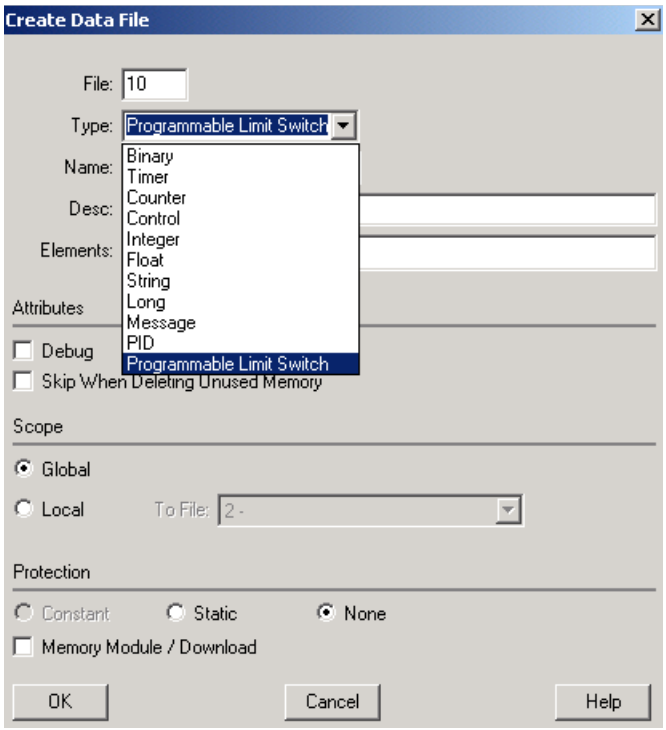
1. Erstellen Sie in RSLogix 500 ein neues Projekt, benennen Sie das Projekt, und wählen Sie die entsprechende Steuerung aus.

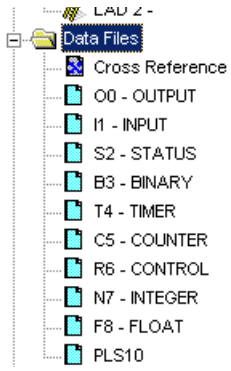



2. Klicken Sie mit der rechten Maustaste auf den Eintrag *Data Files* (Datenfiles), und wählen Sie aus dem Kontextmenü die Option *New* (Neu).



**3.** Geben Sie eine Filenummer ein (9 bis 255), und wählen Sie als Typ *Programmable Limit Switch* (Programmierbarer Endschalter). Sie können gegebenenfalls auch einen Namen und/oder eine Beschreibung eingeben.



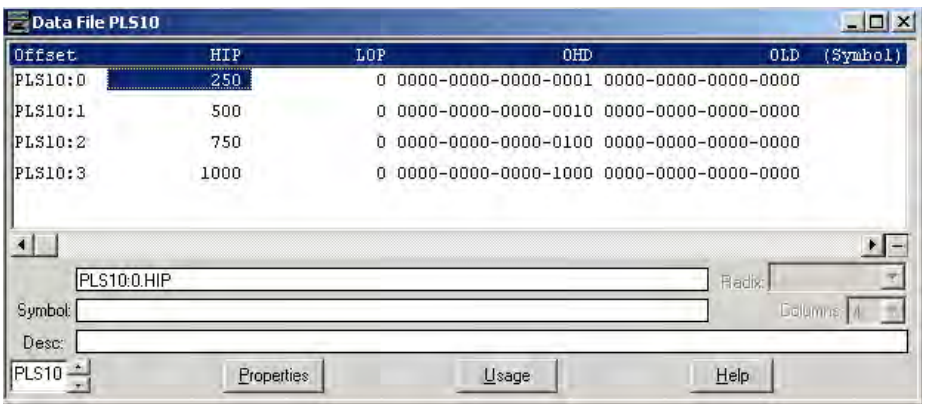




**4.** Die Option *Elements* (Elemente) bezieht sich auf die Anzahl an PLS-Schritten. Wählen Sie für dieses Beispiel den Wert 4.

Wenn Sie zu einem späteren Zeitpunkt weitere Schritte benötigen, gehen Sie zu den Einstellungen für den PLS-Datenfile und erhöhen Sie dort die Anzahl der Elemente.

**5.** Unter *Data Files* sollte der Eintrag *PLS10* angezeigt werden (siehe Abbildung links).

**6.** Doppelklicken Sie unter *Data Files* auf *PLS10*. Geben Sie für dieses Beispiel die unten genannten Werte ein.



Definitionen für PLS-Datenfiles:

| Daten | Beschreibung | Datenformat |
|---|---|---|
| HIP | Oberer Sollwert | 32-Bit mit ganzzahligen Vorzeichen |
| LOP | Unterer Sollwert | |
| OHD | Obere Ausgangsdaten | 16-Bit Binärsystem (Bit 15--> 0000 0000 0000 0000 <--Bit 0) |
| OLD | Untere Ausgangsdaten | |

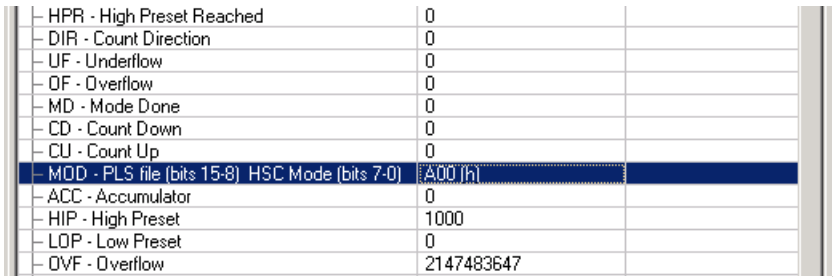
Wenn die Werte für HIP und OHD eingegeben wurden, ist der PLS konfiguriert.

### *Konfigurieren des HSCs für den Gebrauch mit dem PLS*

1. Doppelklicken Sie unter *Controller* (Steuerung) auf *Function Files* (Funktionsfiles).
2. Konfigurieren Sie den HSC.MOD für *HSC:0* so, dass PLS10 verwendet wird und der HSC in Modus 00 läuft.

**WICHTIG** Der Wert für MOD muss als Hexadezimalwert eingegeben werden.

Beispiel: PLS10 = 0A und HSC-Modus = 00


HPR - High Preset Reached 0
DIR - Count Direction 0
UF - Underflow 0
OF - Overflow 0
MD - Mode Done 0
CD - Count Down 0
CU - Count Up 0
MOD - PLS file (bits 15-8) HSC Mode (bits 7-0) A00 [h]
ACC - Accumulator 0
HIP - High Preset 1000
LOP - Low Preset 0
OVF - Overflow 2147483647

### *PLS-Betrieb in diesem Beispiel*

Wenn die Kontaktplanlogik das erste Mal ausgeführt wird, ist HSC.ACC gleich 0 und somit werden die Daten für PLS10:0.OLD durch die HSC.OMB-Maske gesendet und alle Ausgänge geschlossen.

Wenn HSC.ACC den Wert 250 annimmt, wird PLS10:0.OHD durch die HSC.OMB-Maske gesendet und die Ausgänge werden erregt.

Dies wiederholt sich, sobald HSC.ACC die Werte 500, 750 und 1000 erreicht. Nach Abschluss wird der Zyklus zurückgesetzt und beginnt von neuem.

# Verwenden von Hochgeschwindigkeitsausgängen

Mit den Befehlen für Hochgeschwindigkeitsausgänge können Sie die PTO- und die PWM-Funktion steuern und überwachen; diese Funktionen steuern die physischen Hochgeschwindigkeitsausgänge.

| Befehl | Zweck | Seite |
|---|---|---|
| PTO - Impulsausgang | Erzeugung von Schrittimpulsen | 6-2 |
| PWM - Pulsweitenmodulation | Erzeugung eines PWM-Ausgangs | 6-19 |

## PTO – Pulse Train Output (Impulsausgang)

**WICHTIG** Die PTO-Funktion kann nur mit den eingebetteten E/A der Steuerung verwendet werden. Sie kann nicht mit Erweiterungs-E/A-Modulen verwendet werden.

**WICHTIG** Der PTO-Befehl darf nur mit den MicroLogix 1200- und 1500 BXB-Geräten verwendet werden. Die Relaisaus- gänge sind für den Hochgeschwindigkeitsbetrieb nicht geeignet.

Befehlstyp: Ausgang

**Tabelle 6.1 Ausführungszeit des PTO-Befehls**

| Steuerung | Strompfad: | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 75,6 µs | 24,4 µs |
| MicroLogix 1500 | 72,6 µs | 21,1 µs |

## PTO-Funktion (Impulsausgang)

Die MicroLogix 1200 1762-L24BXB- und 1762-L40BXB-Steuerungen unterstützen jeweils einen Hochgeschwindigkeitsausgang. Eine MicroLogix 1500-Steuerung, die ein 1764-28BXB-Basisgerät ver- wendet, unterstützt zwei Hochgeschwindigkeitsausgänge. Diese Ausgänge können als Standardausgänge (nicht Hochgeschwindig- keitsausgänge) verwendet werden. Sie lassen sich auch für den PTO- oder PWM-Betrieb einzeln konfigurieren. Die PTO-Funktion ermög- licht die Erzeugung eines einfachen Bewegungs- oder Impulsprofils direkt durch die Steuerung. Das Impulsprofil umfasst drei Hauptkomponenten:

- Anzahl der zu erzeugenden Impulse
- Beschleunigungs-/Verzögerungsintervalle
- Ausführungsintervall

Der PTO-Befehl unterscheidet sich, wie auch die HSC- und die PWM-Funktion, von den meisten anderen Steuerungsbefehlen. Diese Befehle werden von speziellen Schaltungen ausgeführt, die parallel zum Hauptsystemprozessor aktiv sind. Dieser Aufbau ist aufgrund der Hochleistungsanforderungen dieser Funktionen erforderlich.

Bei dem vorliegenden Anwendungsfall legt der Anwender die Anzahl der insgesamt zu erzeugenden Impulse (die der zurückzulegenden Strecke entsprechen) sowie die Anzahl der für jede Beschleunigungs-/ Verzögerungsperiode zu verwendenden Impulse fest. Über die Anzahl der Impulse, die nicht in der Beschleunigungs-/Verzögerungsperiode verwendet werden, wird auch die Anzahl der Impulse festgelegt, die während der Ausführungsphase erzeugt werden. Bei dem vorliegen- den Anwendungsfall sind das Beschleunigungs- und das Verzögerungs- intervall identisch.

Innerhalb des PTO-Funktionsfiles befinden sich PTO-Elemente. Ein Element kann so definiert werden, dass entweder Ausgang 2 (O0:0/2 bei 1762-L24BXB, 1762-L40BXB und 1764-28BXB) oder Ausgang 3 (O0:0/3 nur bei 1764-28BXB) gesteuert wird.

Die Schnittstelle zu dem PTO-Subsystem wird durch Abfrage eines PTO-Befehls in dem Hauptprogrammfile (Filenummer 2) oder durch Abfrage eines PTO-Befehls in einem beliebigen Unterprogrammfile realisiert. Das folgende Beispiel zeigt den typischen Arbeitsablauf eines PTO-Befehls:

1. Der Strompfad, auf dem sich der PTO-Befehl befindet, wird als wahr erkannt.
2. Der PTO-Befehl wird gestartet, und Impulse werden entsprechend der Beschleunigungs-/Verzögerungsparameter (ACCEL) erzeugt, über die die Anzahl der ACCEL-Impulse und der Profiltyp definiert werden: S-Kurve oder trapezförmig.
3. Die Beschleunigungsphase (ACCEL) wird abgeschlossen.
4. Die Durchführungsphase (RUN) beginnt, und die Anzahl der für diese Phase definierten Impulse werden erzeugt.
5. Die Durchführungsphase (RUN) wird abgeschlossen.

6. Die Verzögerungsphase (DECEL) beginnt, und Impulse werden entsprechend der Beschleunigungs-/Verzögerungsparameter erzeugt, über die die Anzahl der DECEL-Impulse und der Profiltyp definiert werden: S-Kurve oder trapezförmig.
7. Die Verzögerungsphase (DECEL) wird abgeschlossen.
8. Der PTO-Befehl ist abgeschlossen (DONE).

Während der Ausführung des PTO-Befehls werden Status-Bits und -informationen bei laufendem Betrieb der Hauptsteuerung aktualisiert. Da der PTO-Befehl von einem Parallelsystem ausgeführt wird, werden die Status-Bits und anderen Informationen jedes Mal aktualisiert, wenn der PTO-Befehl während der Ausführung abgefragt wird. Auf diese Weise hat das Steuerprogramm während der Ausführung Zugriff auf den PTO-Status.

**HINWEIS** Der PTO-Status kann immer nur so aktuell sein wie die Abfragezeit der Steuerung. Die längste Latenzzeit entspricht der maximalen Abfragezeit der Steuerung. Dieser Effekt kann durch Verwendung eines PTO- Befehls in dem STI-File (wählbarer zeitgesteuerter Interrupt) oder durch Einfügen der PTO-Befehle in das Programm minimiert werden, da auf diese Weise die Anzahl der Abfragen eines PTO-Befehls erhöht wird.

Die in den folgenden Beispielen verwendeten Diagramme zeigen das typische zeitabhängige Ablaufverhalten eines PTO-Befehls. Die in jedem Diagramm dargestellten Phasen stehen in keinerlei Zusammen- hang mit der Abfragezeit der Steuerung. Sie stellen nur ein Sequenz von Ereignissen dar. In der Praxis führt die Steuerung unter Umständen in jeder der in den Beispielen dargestellten Phasen Hunderte oder Tausende Abfragen durch.

### *Bedingungen für den Start des PTO-Befehls*

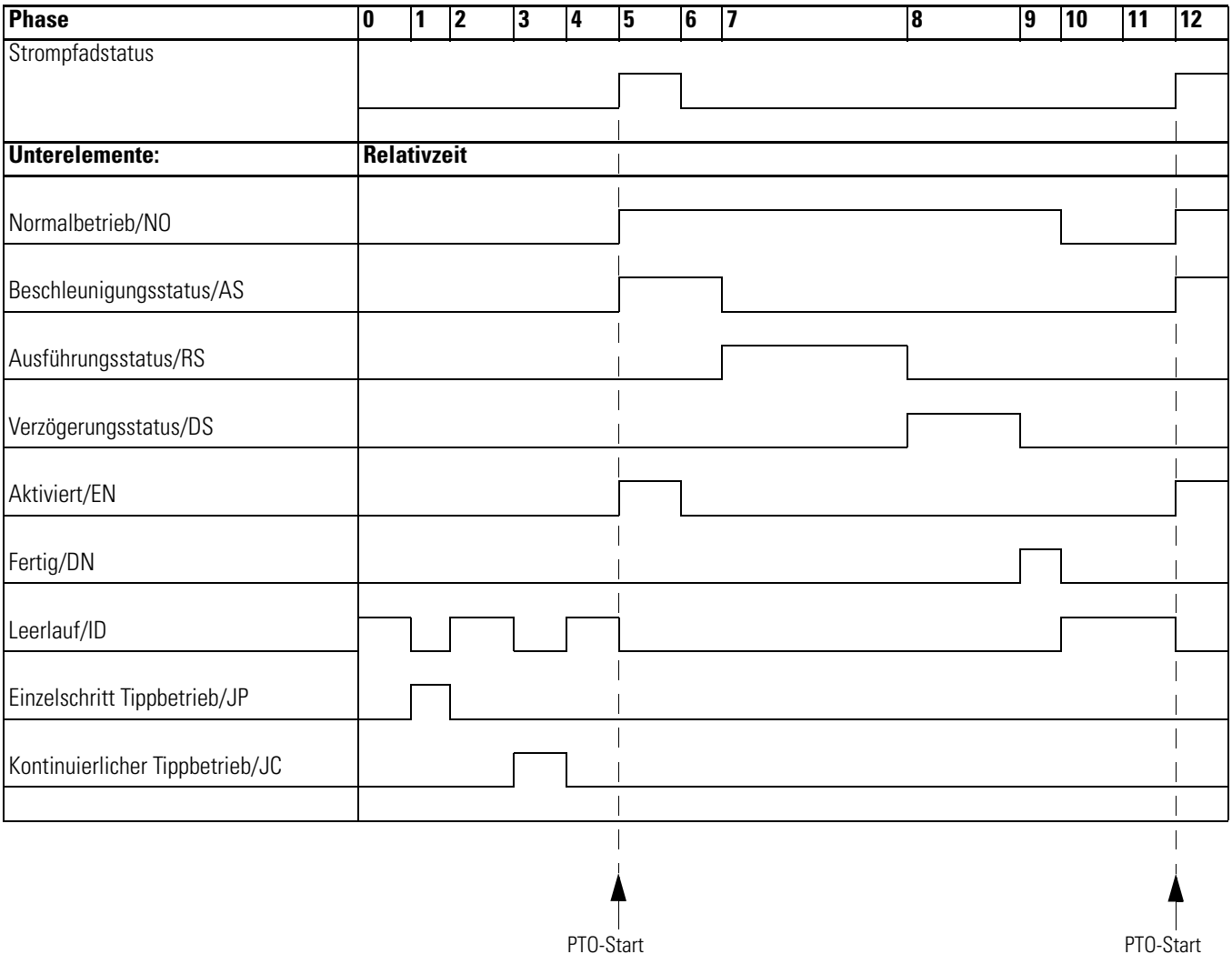
Der PTO-Befehl kann nur gestartet werden, wenn folgende Bedingungen erfüllt sind:

- Der PTO-Befehl muss sich in einem Leerlaufzustand befinden.
- Der Leerlaufzustand ist durch folgende Bedingungen gekennzeichnet:
  - Das JP-Bit (Einzelschritt Tippbetrieb) muss deaktiviert sein.
  - Das JC-Bit (kontinuierlicher Tippbetrieb) muss deaktiviert sein.
  - Das EH-Bit (Hardstopp aktivieren) muss deaktiviert sein.
  - Das NS-Bit (Normalbetrieb) muss deaktiviert sein.
  - Der Ausgang darf nicht forciert werden.
- Der Strompfad befindet sich in einem Übergang von einem unwahren Zustand (0) zu einem wahren Zustand (1).

## Beispiel für Standardaktivierung der Logik

In diesem Beispiel hat der Strompfad den Status eines Übergangsein- gangs. Das bedeutet, dass der PTO-Befehl durch den Übergang des Strompfads von unwahr zu wahr aktiviert wird und der Strompfad noch vor Abschluss des PTO-Befehls wieder einen unwahren Status annimmt.

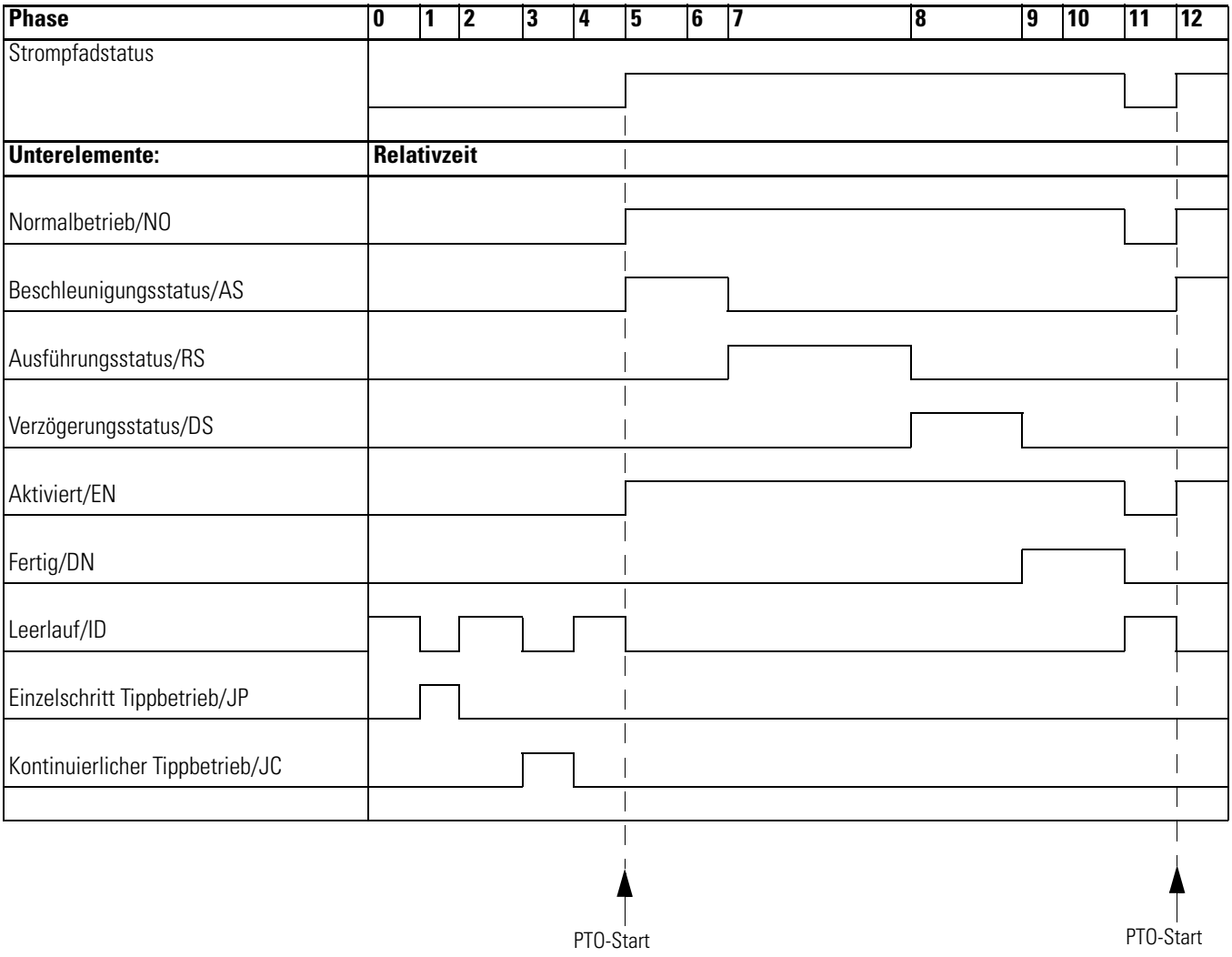
Wenn für den PTO-Befehl ein Übergangseingang verwendet wird, wird bei Abschluss des Befehls das Fertig-Bit (DN) gesetzt, das jedoch nur bis zur nächsten Abfrage des PTO-Befehls durch das Anwender- programm gesetzt bleibt. Über die Struktur des Steuerprogramms ist festgelegt, wann das DN-Bit ausgeschaltet wird. Sie können also durch Überwachung der Status-Bits Fertig (DN), Leerlauf (ID) oder Normal- betrieb (NO) feststellen, wann der Ausgang des PTO-Befehl abgeschlossen ist.



## Beispiel für Standardaktivierung der Logik

In diesem Beispiel hat der Strompfad den Status eines Dauereingangs. Das bedeutet, der Strompfad aktiviert den Normalbetrieb (NO) des PTO-Befehls und hält diesen Logikstatus bis zum Abschluss des PTO-Befehls aufrecht. Bei diesem Logiktyp weist das Status-Bit folgendes Verhalten auf:

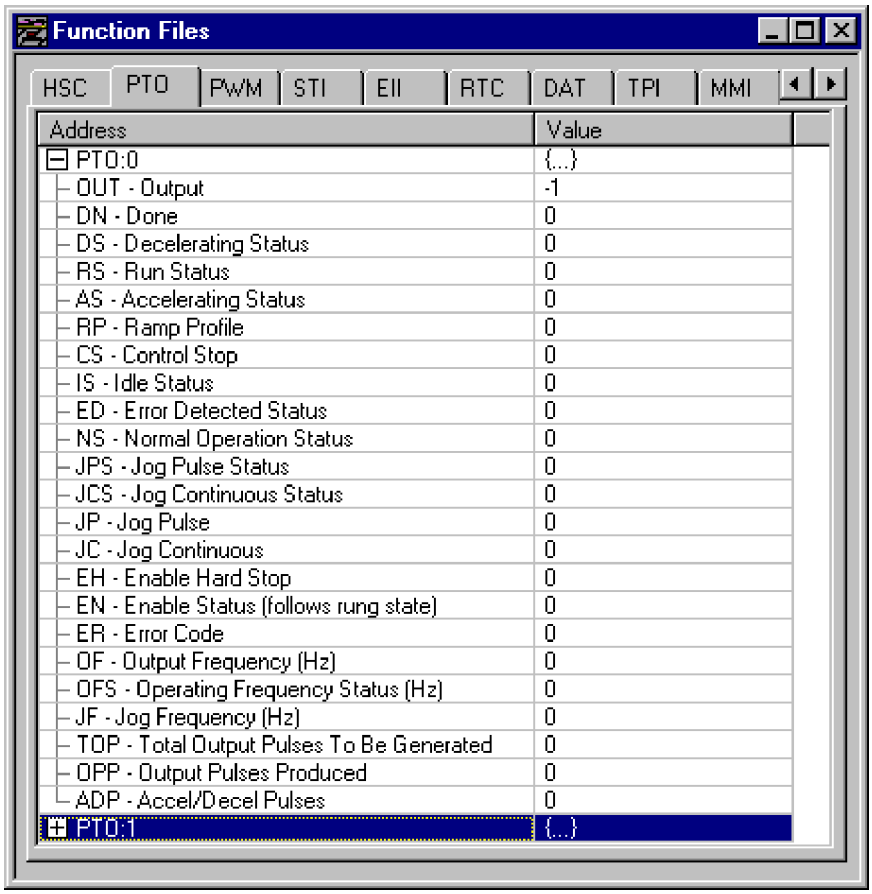
Das Fertig-Bit (DN) wird nach Abschluss des PTO-Befehls auf wahr gesetzt (1) und bleibt gesetzt, bis die PTO-Strompfadlogik unwahr wird. Die unwahre Strompfadlogik aktiviert den PTO-Befehl erneut. Sie können also durch Überwachung des Fertig-Bits (DN) feststellen, wann der Ausgang des PTO-Befehl abgeschlossen ist.



## PTO-Funktionsfile (Impulsausgang)

Innerhalb des RSLogix 500-Funktionsfileordners befindet sich ein PTO-Funktionsfile mit zwei Elementen, PTO0 (1762-L24BXB, 1762-L40BXB und 1764-28BXB) und PTO1 (nur 1764-28BXB). Diese Elemente bieten Zugriff auf PTO-Konfigurationsdaten und ermöglichen auch den Zugriff des Steuerungsprogramms auf alle Informationen, die sich auf die einzelnen Impulsfolgenausgänge beziehen.

**HINWEIS** Wenn sich die Steuerung im Run-Modus befindet, ändern sich unter Umständen die Daten in den Unterelement-Feldern.



## Zusammenfassung der Unterelemente des PTO-Files

Die Variablen der einzelnen PTO-Unterelemente sowie deren Verhalten und die Zugriffsart des Steuerprogramms auf diese Variablen sind nachfolgend einzeln aufgeführt. Alle Beispiele zeigen PTO 0. PTO 1 (nur MicroLogix 1500) weist jedoch exakt diesselben Begriffe und dasselbe Verhalten auf.

**Tabelle 6.2 Funktionsfile PTO (PTO:0)**

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff | Weitere Informationen |
|---|---|---|---|---|---|---|
| OUT - Ausgang | PTO:0.OUT | Wort (INT) | 2 oder 3 | Steuerung | Nur Lesen | 6-8 |
| DN - PID fertig | PTO:0/DN | Bit | 0 oder 1 | Status | Nur Lesen | 6-8 |
| DS - Verzögerungsstatus | PTO:0/DS | Bit | 0 oder 1 | Status | Nur Lesen | 6-8 |
| RS - Ausführungsstatus | PTO:0/RS | Bit | 0 oder 1 | Status | Nur Lesen | 6-9 |
| AS - Beschleunigungsstatus | PTO:0/AS | Bit | 0 oder 1 | Status | Nur Lesen | 6-9 |
| RP - Rampenprofil | PTO:0/RP | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben | 6-9 |
| IS - Leerlaufstatus | PTO:0/IS | Bit | 0 oder 1 | Status | Nur Lesen | 6-10 |
| ED - Status Fehler erkannt | PTO:0/ED | Bit | 0 oder 1 | Status | Nur Lesen | 6-10 |
| NS - Normalbetriebstatus | PTO:0/NS | Bit | 0 oder 1 | Status | Nur Lesen | 6-16 |
| JPS - Status Einzelschritt Tippbetrieb | PTO:0/JPS | Bit | 0 oder 1 | Status | Nur Lesen | 6-16 |
| JCS - Status kontinuierlicher Tippbetrieb | PTO:0/JCS | Bit | 0 oder 1 | Status | Nur Lesen | 6-17 |
| JP - Einzelschritt Tippbetrieb | PTO:0/JP | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben | 6-16 |
| JC - Kontinuierlicher Tippbetrieb | PTO:0/JC | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben | 6-11 |
| EH - Hard-Stop aktivieren | PTO:0/EH | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben | 6-11 |
| EN - Status aktiv (entsprechend Strompfadstatus) | PTO:0/EN | Bit | 0 oder 1 | Status | Nur Lesen | 6-11 |
| ER - Fehlercode | PTO:0.ER | Wort (INT) | -2 bis 7 | Status | Nur Lesen | 6-18 |
| OF - Ausgangsfrequenz (Hz) | PTO:0.OF | Wort (INT) | 0 bis 20000 | Steuerung | Lesen/Schreiben | 6-12 |
| OFS - Ausgangsfrequenzstatus (Hz) | PTO:0.OFS | Wort (INT) | 0 bis 20000 | Status | Nur Lesen | 6-12 |
| JF - Tipp-Frequenz (Hz) | PTO:0.JF | Wort (INT) | 0 bis 20000 | Steuerung | Lesen/Schreiben | 6-16 |
| TOP - Anzahl der zu erzeugenden Impulse | PTO:0.TOP | Doppelwort (32-Bit INT) | 0 bis 2 147 483 647 | Steuerung | Lesen/Schreiben | 6-12 |
| OPP - Erzeugte Ausgangsimpulse | PTO:0.OPP | Doppelwort (32-Bit INT) | 0 bis 2 147 483 647 | Status | Nur Lesen | 6-13 |
| ADP - Beschleunigungs-/ Verzögerungsimpulse | PTO:0.ADP | Doppelwort (32-Bit INT) | siehe S. 6-13 | Steuerung | Lesen/Schreiben | 6-13 |
| CS - Gesteuerter Halt | PTO:0/CS | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben | 6-15 |

## PTO-Ausgang (OUT)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| OUT - Ausgang | PTO:0.OUT | Wort (INT) | 2 oder 3 | Steuerung | Nur Lesen |

Die PTO-Variable OUT (Ausgang) legt den Ausgang (O0:0/2 oder O0:0/3) fest, der von dem PTO-Befehl gesteuert wird. Diese Variable wird bei der Erstellung des Steuerprogramms in dem Funktionsfile- ordner gesetzt und kann nicht durch das Anwenderprogramm gesetzt werden.

- Ist OUT = 2, taktet PTO Ausgang 2 (O0:0.0/2) der integrierten Ausgänge (1762-L24BXB, 1762-L40BXB und 1764-28BXB).
- Ist OUT = 3, taktet PTO Ausgang 3 (O0:0.0/3) der integrierten Ausgänge (nur 1764-28BXB).

**HINWEIS** Durch Forcen eines durch den PTO gesteuerten Ausgangs während der Ausführung werden alle Ausgangsimpulse gestoppt und ein PTO-Fehler generiert.

## PTO-Fertig (DN)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| DN - PID fertig | PTO:0/DN | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit DN (Fertig) wird durch das PTO-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden. Funktionsweise des DN-Bits:

- Gesetzt (1) - Wenn ein PTO-Befehl erfolgreich abgeschlossen wurde.
- Gelöscht (0) - Wenn der Strompfad, in dem sich der PTO-Befehl befindet, unwahr ist. Wenn der Strompfad bei Abschluss des PTO-Befehls unwahr ist, wird das Fertig-Bit bis zur nächsten Abfrage des PTO-Befehls gesetzt.

## PTO-Verzögerungsstatus (DS)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| DS - Verzögerungsstatus | PTO:0/DS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit DS (Verzögerung) wird durch das PTO-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden. Funktionsweise des DS-Bits:

- Gesetzt (1) - Wenn sich ein PTO-Befehl in der Verzögerungsphase des Ausgangsprofils befindet.
- Gelöscht (0) - Wenn sich ein PTO-Befehl nicht in der Verzögerungsphase des Ausgangsprofils befindet.

## PTO-Ausführungsstatus (RS)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| RS - Ausführungs-status | PTO:0/RS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit RS (Ausführungsstatus) wird durch das PTO-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad inner- halb des Steuerprogramms verwendet werden. Funktionsweise des RS-Bits:

- Gesetzt (1) - Wenn sich ein PTO-Befehl in der Ausführungsphase des Ausgangsprofils befindet.
- Gelöscht (0) - Wenn sich ein PTO-Befehl nicht in der Ausfüh- rungsphase des Ausgangsprofils befindet.

## PTO-Beschleunigungsstatus (AS)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| AS - Beschleunigungs-status | PTO:0/AS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit AS (Beschleunigungsstatus) wird durch das PTO- Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden. Funktionsweise des AS-Bits:

- Gesetzt (1) - Wenn sich ein PTO-Befehl in der Beschleunigungsphase des Ausgangsprofils befindet.
- Gelöscht (0) - Wenn sich ein PTO-Befehl nicht in der Beschleunigungsphase des Ausgangsprofils befindet.

## PTO-Rampenprofil (RP)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| RP - Rampenprofil | PTO:0/RP | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben |

Das PTO-Bit RP (Rampenprofil) steuert die Beschleunigungs- und Verzögerungsrate der durch das PTO-Subsystem erzeugten Ausgangsimpulse bis zu/von der Ausgangsfrequenz, die in dem PTO-Funktionsfile (PTO:0.OF) festgelegt ist. Es kann von jedem Eingangs- oder Ausgangsbefehl in jedem Strompfad innerhalb des Steuerprogramms verwendet werden. Funktionsweise des RP-Bits:

- Gesetzt (1) - Der PTO-Befehl erzeugt ein S-Kurven-Profil.
- Gelöscht (0) - Der PTO-Befehl erzeugt ein trapezförmiges Profil.

## PTO-Leerlaufstatus (IS)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| IS - Leerlaufstatus | PTO:0/IS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit IS (Leerlaufstatus) wird durch das PTO-Subsystem gesteuert. Es kann von Eingangsbefehlen in dem Steuerprogramm verwendet werden. Das PTO-Subsystem muss sich in einem Leerlauf- status befinden, wenn ein PTO-Betrieb gestartet werden soll.

Funktionsweise des IS-Bits:

- Gesetzt (1) - Das PTO-Subsystem befindet sich in einem Leerlauf- status. Der Leerlaufstatus ist wie folgt definiert: Der PTO wird nicht ausgeführt, und es sind keine Fehler vorhanden.
- Gelöscht (0) - Das PTO-Subsystem befindet sich nicht in einem Leerlaufstatus (es wird ausgeführt).

## PTO-Fehler erkannt (ED)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| ED - Status Fehler erkannt | PTO:0/ED | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit ED (Status Fehler erkannt) wird durch das PTO-Sub- system gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden, um festzustellen, wenn sich der PTO-Befehl in einem Fehlerzustand befindet. Wenn ein Fehlerzustand erkannt wird, wird der betreffende Fehler in dem Fehlercoderegister (PTO:0.ER) gekennzeichnet. Funktionsweise des DN-Bits:

- Gesetzt (1) - Wenn sich ein PTO-Befehl in einem Fehlerzustand befindet.
- Gelöscht (0) - Wenn sich ein PTO-Befehl nicht in einem Fehlerzustand befindet.

## PTO-Normalbetriebsstatus (NS)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| NS - Normalbetriebstatus | PTO:0/NS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit NS (Normalbetriebstatus) wird durch das PTO-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad inner- halb des Steuerprogramms verwendet werden, um festzustellen, wenn sich der PTO-Befehl in seinem normalen Betriebsstatus befindet. Zu den normalen Betriebszuständen gehören: ACCEL (Beschleunigung), RUN (Ausführung), DECEL (Verzögerung) und DONE (Fertig), sofern dabei keine PTO-Fehler vorliegen. Funktionsweise des NS-Bits:

- Gesetzt (1) - Wenn sich ein PTO-Befehl in seinem Normalzustand befindet.
- Gelöscht (0) - Wenn sich ein PTO-Befehl nicht in seinem Normalzustand befindet.

## PTO-Hard-Stop aktivieren (EH)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| EH - Hard-Stop aktivieren | PTO:0/EH | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben |

Mit dem PTO-Bit EH (Hard-Stop aktivieren) kann das PTO-Subsystem sofort angehalten werden. Sobald das PTO-Subsystem eine Impuls- sequenz gestartet hat, ist die Aktivierung des EH-Bits die einzige Möglichkeit, die Erzeugung von Impulsen zu stoppen. Das EH-Bit bricht jeglichen Betrieb des PTO-Subsystems (Leerlauf, Normalbetrieb, Kontinuierliche Impulse erzeugen oder Impulse erzeugen) ab und erzeugt einen PTO-Subsystemfehler. Funktionsweise des EH-Bits:

- Gesetzt (1) - Das PTO-Subsystem wird angewiesen, die Erzeugung von Impulsen sofort anzuhalten (Ausgang aus = 0).
- Gelöscht (0) - Normalbetrieb

## PTO-Status aktiv (EN)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| EN - Status aktiv (entsprechend Strompfadstatus) | PTO:0/EN | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit EN (Status Aktiv) wird durch das PTO-Subsystem ge- steuert. Wenn der Strompfad vor dem PTO-Befehl wahr ist, wird der PTO-Befehl aktiviert und das Status-Bit Aktiv gesetzt. Wenn der Strompfad vor dem PTO-Befehl vor Abschluss der Impulssequenz unwahr wird, wird das Status-Bit Aktiv rückgesetzt (0). Funktionsweise des EN-Bits:

- Gesetzt (1) - PTO aktiv
- Gelöscht (0) - PTO abgeschlossen oder Strompfad vor PTO ist unwahr

## PTO-Ausgangsfrequenz (OF)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| OF - Ausgangsfrequenz (Hz) | PTO:0.OF | Wort (INT) | 0 bis 20000 | Steuerung | Lesen/Schreiben |

Die PTO-Variable OF (Ausgangsfrequenz) definiert die Frequenz des PTO-Ausgangs während der Ausführungsphase des Impulsprofils. Dieser Wert wird in der Regel bestimmt durch den Gerätetyp, der gesteuert wird, die Mechanik der Anwendung oder das/die zu bewegende/n Gerät/Komponenten. Bei Werten kleiner als null oder größer als 20.000 wird ein PTO-Fehler erzeugt.

## PTO-Ausgangsfrequenzstatus (OFS)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| OFS - Ausgangsfrequenzstatus (Hz) | PTO:0.OFS | Wort (INT) | 0 bis 20000 | Status | Nur Lesen |

Der PTO-OFS (Ausgangsfrequenzstatus) wird durch das PTO-Subsystem generiert und kann in dem Steuerprogramm zur Überwachung der tatsächlichen Frequenz verwendet werden, die durch das PTO-Subsystem erzeugt wird.

**HINWEIS** Der angezeigte Wert ist unter Umständen nicht vollständig mit dem Wert identisch, der unter PTO:0.OF eingegeben wurde. Dieser Unterschied ist darauf zurückzuführen, dass das PTO-Subsystem unter Umständen nicht in der Lage ist, bei höheren Frequenzen eine exakte Frequenz einzustellen. Bei PTO-Anwendungen ist dies in der Regel nicht relevant, da in allen Fällen eine genaue Anzahl an Impulsen erzeugt wird.

## PTO-Anzahl der zu erzeugenden Impulse (TOP)

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| TOP - Anzahl der zu erzeugenden Impulse | PTO:0.TOP | Doppelwort (32-Bit INT) | 0 bis 2 147 483 647 | Steuerung | Lesen/Schreiben |

Der PTO-TOP (Total Output Pulses; Anzahl der zu erzeugenden Impulse) gibt die Anzahl der Impulse an, die für das Impulsprofil erzeugt werden müssen (für die Phasen *Beschleunigung/Ausführung/Verzögerung*).

## PTO-erzeugte Ausgangsimpulse (OPP)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| OPP - Erzeugte Ausgangsimpulse | PTO:0.OPP | Doppelwort (32-Bit INT) | 0 bis 2 147 483 647 | Status | Nur Lesen |

Der PTO-OPP (Erzeugte Ausgangsimpulse) wird durch das PTO-Subsystem generiert und kann in dem Steuerprogramm zur Überwachung der durch das PTO-Subsystem erzeugten Impulse verwendet werden.

## PTO-Beschleunigungs-/Verzögerungsimpulse (ADP)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| ADP - Beschleunigungs-/ Verzögerungsimpulse | PTO:0.ADP | Doppelwort (32-Bit INT) | siehe unten | Steuerung | Lesen/Schreiben |

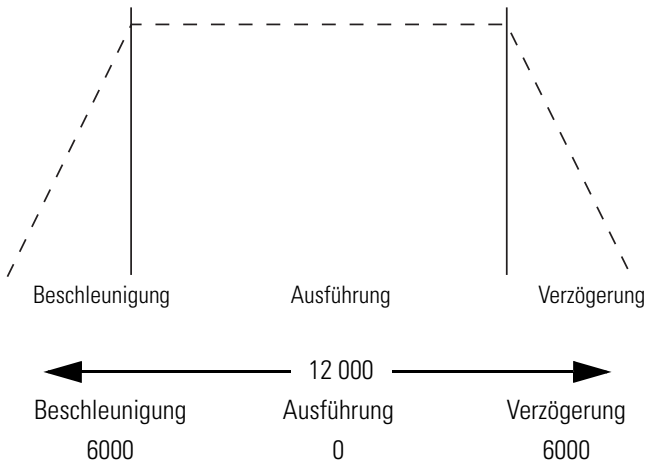
Der PTO ADP (Beschleunigungs-/Verzögerungsimpulse) legt die Anzahl der Impulse (TOP-Variable) fest, die in der Beschleunigungs- und der Verzögerungsphase erzeugt werden sollen. Der ADP be- stimmt die Beschleunigungs- und Verzögerungsrate von 0 auf die PTO-Ausgangsfrequenz (OF). Die PTO-Ausgangsfrequenz (OF) bestimmt die Betriebsfrequenz in Impulsen/Sekunde während der Ausführungsphase des Profils.

**HINWEIS** Nachdem die ADP-Parameter eingegeben wurden, erzeugt das PTO einen Beschleunigungs-/Verzögerungsfehler, wenn einer der folgenden Zustände eintritt:

- Der ADP-Wert ist kleiner als 0.
- Der ADP-Wert ist größer als die Hälfte aller zu erzeugenden Ausgangsimpulse (TOP).

Für das folgende Beispiel gilt:

- TOP (Anzahl zu erzeugender Ausgangsimpulse) = 12 000
- ADP (Beschleunigungs-/Verzögerungsimpulse) = 6 000 (Dies ist der ADP-Maximalwert, der angenommen werden kann, ohne einen Fehler hervorzurufen. In der Ausführungsphase beträgt der Wert 0.)




Im vorliegenden Beispiel könnte für die Beschleunigung/Verzögerung ein Wert von maximal 6000 verwendet werden, denn wenn sowohl die Beschleunigungs- als auch die Verzögerungsphase 6000 Impulse umfassen, ist die Summe der Impulse = 12 000. Die Ausführungs- komponente wäre in diesem Fall Null. Dieses Profil würde aus einer Beschleunigungsphase von 0 bis 6000 bestehen. Bei 6000 wird die Ausgangsfrequenz (OF-Variable) erzeugt, und unmittelbar danach beginnt die Verzögerungsphase von 6000 bis 12 000. Bei 12 000 wird der PTO-Betrieb beendet (Ausgangsfrequenz = 0).

Bestimmung der Rampendauer (Beschleunigungs-/Verzögerungsrampe):

- 2 x ADP/OF = Dauer in Sekunden (OF = Ausgangsfrequenz)

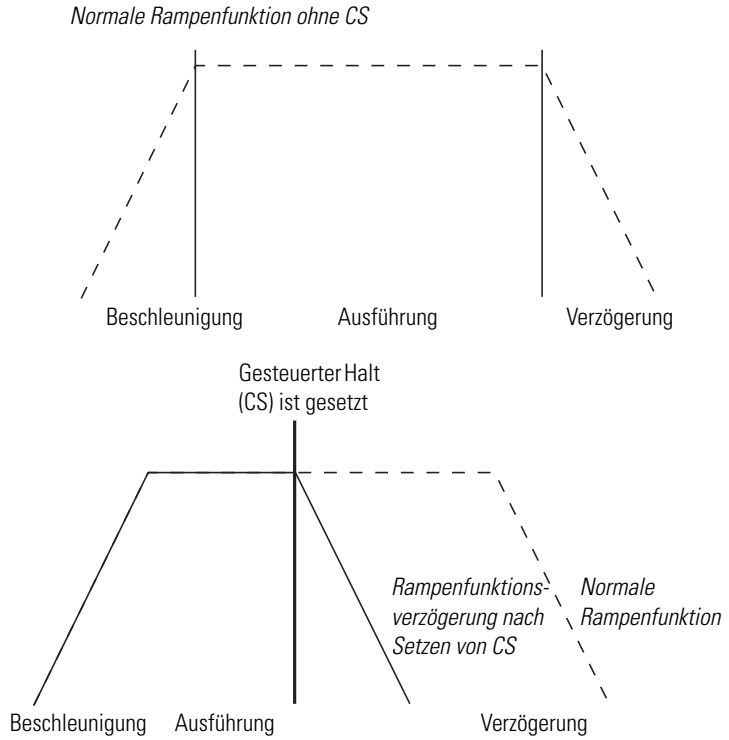
Anhand der folgenden Formeln kann die maximale Frequenz für beide Profile berechnet werden. Maximalfrequenz = Ganzzahl, die kleiner als das oder gleich dem unten genannten Ergebnis ist (OF = Ausgangsfrequenz):

- Trapezförmige Profile: [OF x (OF/4)] + 0,5
- S-Kurven-Profile: 0,999 x OF x SQRT(OF/6)

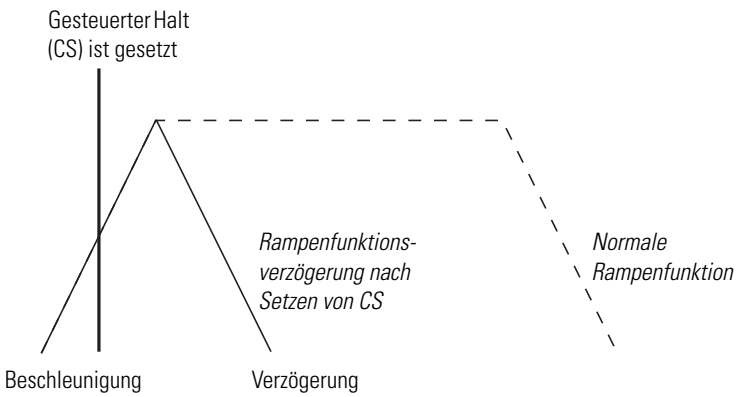
## PTO-gesteuerter Halt (CS)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| CS - Gesteuerter Halt | PTO:0/CS | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben |

Das PTO-Bit CS (Controlled Stop) wird zum Anhalten eines aus- führenden PTO-Befehls im Ausführungsteil des Profils verwendet, indem die Verzögerungsphase sofort gestartet wird. Durch Setzen dieses Bits wird die Verzögerungsphase ohne Fehler oder Fehler- zustand abgeschlossen.




Wird das CS-Bit während der Beschleunigungsphase gesetzt, wird die Beschleunigungsphase abgeschlossen und das PTO geht sofort in die Verzögerungsphase über.



## PTO-Tipp-Frequenz (JF)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| JF - Tipp-Frequenz (Hz) | PTO:0.JF | Wort (INT) | 0 bis 20000 | Steuerung | Lesen/Schreiben |

Die PTO-Variable JF (Tipp-Frequenz) definiert die Frequenz des PTO-Ausgangs während aller Tippbetriebs-Phasen. Dieser Wert wird in der Regel bestimmt durch den Gerätetyp, der gesteuert wird, die Mechanik der Anwendung oder das/die zu bewegende/n Gerät/Komponenten. Bei Werten kleiner als null oder größer als 20.000 wird ein PTO-Fehler erzeugt.

## PTO – Einzelschritt Tippbetrieb (JP)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| JP - Einzelschritt Tippbetrieb | PTO:0/JP | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben |

Über das PTO-Bit JP (Einzelschritt Tippbetrieb) wird das PTO-Sub- system angewiesen, einen einzelnen Impuls zu erzeugen. Die Impulsdauer wird durch den JF-Parameter (Tipp-Frequenz) in dem PTO-Funktionsfile bestimmt. Der JP-Betrieb ist nur möglich, wenn folgende Bedingungen erfüllt sind:

- PTO-Subsystem im Leerlauf
- Kontinuierlicher Tippbetrieb nicht aktiv
- Aktiv-Bit (EN) ausgeschaltet

Funktionsweise des JP-Bits:

- Gesetzt (1) - Das PTO-Subsystem wird angewiesen, einen einzelnen Impuls zu erzeugen
- Gelöscht (0) - Das PTO-Subsystem JP (Einzelschritt Tippbetrieb) wird vorbereitet

## PTO – Status Einzelschritt Tippbetrieb (JPS)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| JPS - Status Einzelschritt Tippbetrieb | PTO:0/JPS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit JPS (Status Einzelschritt Tippbetrieb) wird durch das PTO-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden, um festzustellen, wenn der PTO-Befehl einen einzelnen Impuls erzeugt.

Funktionsweise des JPS-Bits:

- Gesetzt (1) - Wenn ein PTO-Befehl einen einzelnen Impuls ausgibt
- Gelöscht (0) - Wenn ein PTO-Befehl den Status Einzelschritt Tippbetrieb verlässt

**HINWEIS** Bei Abschluss des Ausgangsimpuls bleibt das JP-Bit in der Regel gesetzt. Das JPS-Bit bleibt gesetzt, bis das JP-Bit gelöscht wird (0 = aus).

## PTO – Kontinuierlicher Tippbetrieb (JC)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| JC - Kontinuierlicher Tippbetrieb | PTO:0/JC | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben |

Über das PTO-Bit JC (Kontinuierlicher Tippbetrieb) wird das PTO-Subsystem angewiesen, kontinuierlich Impulse zu erzeugen. Die erzeugte Frequenz wird durch den JF-Parameter (Tipp-Frequenz) in dem PTO-Funktionsfile bestimmt. Der JC-Betrieb ist nur möglich, wenn folgende Bedingungen erfüllt sind:

- PTO-Subsystem im Leerlauf
- Einzelschritt Tippbetrieb nicht aktiv
- Aktiv-Bit (EN) ausgeschaltet

Funktionsweise des JC-Bits:

- Gesetzt (1) - Das PTO-Subsystem wird angewiesen, kontinuierlich Impulse zu erzeugen
- Gelöscht (0) - Das PTO-Subsystem erzeugt keine Impulse

Wenn das JC-Bit gelöscht wird, wird der aktuelle Ausgangsimpuls abgebrochen.

## PTO – Status Kontinuierlicher Tippbetrieb (JCS)

| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| JCS - Status kontinuierlicher Tippbetrieb | PTO:0/JCS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PTO-Bit JCS (Status Kontinuierlicher Tippbetrieb) wird durch das PTO-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden, um festzustellen, wenn die PTO-Funktion kontinuierlich Impulse erzeugt. Funktionsweise des JCS-Bits:

- Gesetzt (1) - Das PTO-Subsystem erzeugt kontinuierlich Impulse
- Gelöscht (0) - Das PTO-Subsystem erzeugt keine Impulse

## PTO-Fehlercode (ER)

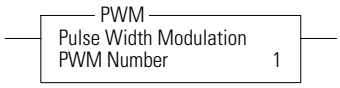
| Beschreibung Unterelement | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| ER - Fehlercode | PTO:0.ER | Wort (INT) | -2 bis 7 | Status | Nur Lesen |

PTO-Fehlercodes, die von dem PTO-Subsystem erkannt wurden, werden in diesem Register angezeigt. Die Fehlercodes werden in der nachfolgenden Tabelle beschrieben.

**Tabelle 6.3 PTO-Fehlercodes**

| Fehler-code | Nicht vom Anwender verursach-ter Fehler | Korrigier-barer Fehler | Befehls-fehler | Fehler-name | Beschreibung |
|---|---|---|---|---|---|
| -2 | Ja | Nein | Nein | Über-lappungs-fehler | Eine Ausgangsüberlappung wurde festgestellt. Mehrere Funktionen wurden demselben physischen Ausgang zugewiesen. Dies ist ein Konfigurations-fehler. Ein Steuerungsfehler wird erzeugt, und die Anwenderfehlerroutine wird nicht ausgeführt. Beispiel: PTO0 und PTO1 versuchen, denselben Ausgang zu verwenden. |
| -1 | Ja | Nein | Nein | Ausgangs-fehler | Ein ungültiger Ausgang wurde angegeben. Nur Ausgang 2 und Ausgang 3 stehen zur Verfügung. Dies ist ein Konfigurationsfehler. Ein Steuerungsfehler wird erzeugt, und die Anwenderfehlerroutine wird nicht ausgeführt. |
| 0 | --- | --- | | Normal | Normal (0 = kein Fehler aufgetreten) |
| 1 | Nein | Nein | Ja | Hardstopp festge-stellt | Dieser Fehler wird generiert, wenn ein Hardwarehalt festgestellt wurde. Bei diesem Fehler wird kein Steuerungsfehler erzeugt.<br>Zum Löschen dieses Fehlers den PTO-Befehl an einem unwahren Strompfad scannen und das EH-Bit (Enable Hard Stop) auf 0 zurücksetzen. |
| 2 | Nein | Nein | Ja | Ausgang forciert | Der konfigurierte PTO-Ausgang (2 oder 3) ist forciert. Diese forcierte Bedingung *muss* aufgehoben werden, da sonst der PTO-Befehl nicht ausgeführt werden kann.<br>Bei diesem Fehler wird kein Steuerungsfehler erzeugt. Dieser Fehler wird automatisch gelöscht, sobald die forcierte Bedingung aufgehoben wird. |
| 3 | Nein | Ja | Nein | Frequenz-fehler | Der Wert der Ausgangsfrequenz (OFS) ist kleiner als 0 oder größer als 20 000. Bei diesem Fehler wird ein Steuerungsfehler erzeugt. Dieser Fehler kann durch die Logik innerhalb der Anwenderfehlerroutine gelöscht werden. |
| 4 | Nein | Ja | Nein | Beschleu-nigungs-/ Verzöge-rungsfehler | Die Beschleunigungs-/Verzögerungsparameter (ADP):<br>• sind kleiner als null<br>• sind größer als die Hälfte der gesamten zu erzeugenden Ausgangsimpulse (TOP)<br>• „Accel/Decel" überschreitet den Grenzwert (siehe Seite 6-13.)<br>Bei diesem Fehler wird ein Steuerungsfehler erzeugt. Dieser Fehler kann durch die Logik innerhalb der Anwenderfehlerroutine gelöscht werden. |
| 5 | Nein | Nein | Ja | Fehler bei Einzel-schritt Tipp-betrieb | Der PTO ist im Leerlauf und mindestens zwei der folgenden Bits sind gesetzt:<br>• EN-Bit (Aktiv) gesetzt<br>• JP-Bit (Einzelschritt Tippbetrieb) gesetzt<br>• JC-Bit (Kontinuierlicher Tippbetrieb) gesetzt<br>Bei diesem Fehler wird kein Steuerungsfehler erzeugt. Dieser Fehler wird automatisch gelöscht, sobald die Fehlerbedingung aufgehoben wird. |
| 6 | Nein | Ja | Nein | Tipp-Frequenz-Fehler | Der JF-Wert (Tipp-Frequenz) ist kleiner als 0 oder größer als 20 000. Bei diesem Fehler wird ein Steuerungsfehler erzeugt. Dieser Fehler kann durch die Logik innerhalb der Anwenderfehlerroutine gelöscht werden. |
| 7 | Nein | Ja | Nein | Längen-fehler | Die Summe der zu erzeugenden Ausgangsimpulse (TOP) ist kleiner als null. Bei diesem Fehler wird ein Steuerungsfehler erzeugt. Dieser Fehler kann durch die Logik innerhalb der Anwenderfehlerroutine gelöscht werden. |

# PWM – Pulsweitenmodulation




**WICHTIG** Die PWM-Funktion kann nur mit der eingebetteten E/A der Steuerung verwendet werden. Sie kann nicht mit Erweiterungs-E/A-Modulen verwendet werden.

**WICHTIG** Der PWM-Befehl kann nur mit MicroLogix 1200- und 1500 BXB-Geräten verwendet werden. Die Relais- ausgänge sind für den Hochgeschwindigkeitsbetrieb nicht geeignet.

Befehlstyp: Ausgang

**Tabelle 6.4 Ausführungszeit des PWM-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 126,6 µs | 24,7 µs |
| MicroLogix 1500 | 107,4 µs | 21,1 µs |

## PWM-Funktion

Mit Hilfe der PWM-Funktion kann ein Feldgerät über ein PWM-Signal gesteuert werden. Das PWM-Profil enthält zwei Hauptkomponenten:

- Zu erzeugende Frequenz
- Tastgradintervall

Der PWM-Befehl unterscheidet sich, wie auch die HSC- und die PTO-Funktion, von den meisten anderen Steuerungsbefehlen. Diese Befehle werden von speziellen Schaltungen ausgeführt, die parallel zum Hauptsystemprozessor aktiv sind. Dieser Aufbau ist aufgrund der Hochleistungsanforderungen dieser Funktionen erforderlich.

Die Schnittstelle zu dem PWM-Subsystem wird durch Abfrage eines PWM-Befehls in dem Hauptprogrammfile (Filenummer 2) oder durch Abfrage eines PWM-Befehls in einem beliebigen Unterprogrammfile realisiert. Das folgende Beispiel zeigt den typischen Arbeitsablauf eines PWM-Befehls:

1. Der Strompfad, in dem sich ein PWM-Befehl befindet, wird als wahr erkannt (der PWM wird gestartet).
2. Ein PWM-Signal mit der angegebenen Frequenz wird erzeugt.
3. Die Ausführungsphase ist aktiv. Ein PWM-Signal mit der angegebenen Frequenz und dem angegebenen Tastgrad wird ausgegeben.
4. Der Strompfad, auf dem sich der PTO-Befehl befindet, wird als unwahr erkannt.
5. Der PWM-Befehl befindet sich im Leerlaufzustand.

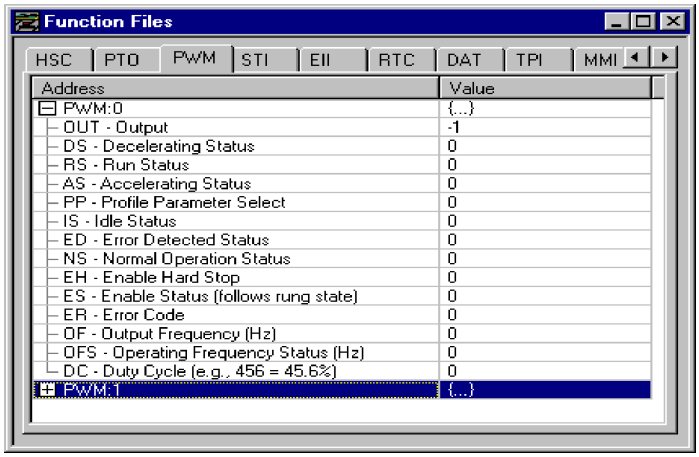
Während der Ausführung des PWM-Befehls werden Status-Bits und -informationen bei laufendem Betrieb der Hauptsteuerung aktualisiert. Da der PWM-Befehl von einem Parallelsystem ausgeführt wird, werden die Status-Bits und andere Informationen jedes Mal aktualisiert, wenn der PWM-Befehl

während der Ausführung abgetastet wird. Auf diese Weise hat das Steuerprogramm während der Ausführung Zugriff auf den PWM-Status.

**HINWEIS** Der PWM-Status kann immer nur so aktuell sein wie die Abfragezeit der Steuerung. Die längste Latenzzeit entspricht der maximalen Abfragezeit der Steuerung. Dieser Effekt kann durch Verwendung eines PWM- Befehls in dem STI-File (Wählbar zeitgesteuerte Interrupts) oder durch Einfügen der PWM-Befehle in das Programm minimiert werden, da auf diese Weise die Anzahl der Abfragen eines PWM-Befehls erhöht wird.

## Funktionsfile für Pulsweitenmodulation (PWM)

Der PWM-Funktionsfile enthält zwei PWM-Elemente. Jedes Element kann entweder zur Steuerung von Ausgang 2 (O0:0/2 für 1762-L24BXB, 1762-L40BXB und 1764-28BXB) oder Ausgang 3 (O0:0/3 nur für 1764-28BXB) verwendet werden. Das Funktionsfile-Element PWM:0 ist nachfolgend abgebildet.



## Zusammenfassung der Elemente des PWM-Files

Die Variablen der einzelnen PWM-Elemente sowie deren Verhalten und die Zugriffsart des Steuerprogramms auf diese Variablen sind nachfolgend einzeln aufgeführt.

| Beschreibung Element | Adresse | Datenformat | Bereich | Typ | Anwenderpro-grammzugriff | Weitere Informationen |
|---|---|---|---|---|---|---|
| OUT - PWM-Ausgang | PWM:0.OUT | Wort (INT) | 2 oder 3 | Status | Nur Lesen | 6-21 |
| DS - Verzögerungsstatus | PWM:0/DS | Bit | 0 oder 1 | Status | Nur Lesen | 6-21 |
| RS - PWM-Ausführungsstatus | PWM:0/RS | Bit | 0 oder 1 | Status | Nur Lesen | 6-22 |
| AS - Beschleunigungsstatus | PWM:0/AS | Bit | 0 oder 1 | Status | Nur Lesen | 6-22 |
| PP - Profilparameterauswahl | PWM:0/PP | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben | 6-23 |
| IS - PWM-Leerlaufstatus | PWM:0/IS | Bit | 0 oder 1 | Status | Nur Lesen | 6-23 |
| ED - PWM-Fehlererkennung | PWM:0/ED | Bit | 0 oder 1 | Status | Nur Lesen | 6-23 |
| NS - PWM-Normalbetrieb | PWM:0/NS | Bit | 0 oder 1 | Status | Nur Lesen | 6-24 |
| EH - PWM Hard-Stop aktivieren | PWM:0/EH | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben | 6-24 |
| ES - PWM Status Aktiv | PWM:0/ES | Bit | 0 oder 1 | Status | Nur Lesen | 6-24 |
| OF - PWM-Ausgangsfrequenz | PWM:0.OF | Wort (INT) | 0 bis 20000 | Steuerung | Lesen/Schreiben | 6-25 |
| OFS - PWM-Betriebsfrequenzstatus | PWM:0.OFS | Wort (INT) | 0 bis 20000 | Status | Nur Lesen | 6-25 |
| DC - PWM-Tastgrad | PWM:0.DC | Wort (INT) | 1 bis 1000 | Steuerung | Lesen/Schreiben | 6-25 |
| DCS - PWM-Tastgradstatus | PWM:0.DCS | Wort (INT) | 1 bis 1000 | Status | Nur Lesen | 6-25 |
| ADD - Beschleunigungs-/ Bremsverzögerung | PWM:0.ADD | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/Schreiben | 6-26 |
| ER - PWM-Fehlercodes | PWM:0.ER | Wort (INT) | -2 bis 5 | Status | Nur Lesen | 6-26 |

### PWM-Ausgang (OUT)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| OUT - PWM-Ausgang | PWM:0.OUT | Wort (INT) | 2 oder 3 | Status | Nur Lesen |

Die PWM-Variable OUT (Output) definiert den physischen Ausgang, den der PWM-Befehl steuert. Diese Variable wird bei der Erstellung des Steuerprogramms in dem Funktionsfileordner gesetzt und kann nicht durch das Anwenderprogramm gesetzt werden. Die Ausgänge werden wie oben aufgeführt als O0:0/2 oder O0:0/3 definiert:

- O0:0.0/2: PWM moduliert Ausgang 2 der integrierten Ausgänge (1762-L24BXB, 1762-L40BXB und 1764-28BXB)
- O0:0.0/3: PWM moduliert Ausgang 3 der integrierten Ausgänge (nur 1764-28BXB)

### PWM-Verzögerungsstatus (DS)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| DS - Verzögerungsstatus | PWM:0/DS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PWM-Bit DS (Verzögerung) wird durch das PWM-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden. Funktionsweise des DS-Bits:

- Gesetzt (1) - Wenn sich ein PWM-Ausgang in der Verzögerungsphase des Ausgangsprofils befindet.
- Gelöscht (0) - Wenn sich ein PWM-Ausgang nicht in der Verzögerungsphase des Ausgangsprofils befindet.

## PWM-Ausführungsstatus (RS)

| Beschreibung Element | Adresse | Datenformat | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| RS - PWM-Ausführungs- status | PWM:0/RS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PWM-Bit RS (Ausführungsstatus) wird durch das PWM-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad inner- halb des Steuerprogramms verwendet werden.

- Gesetzt (1) - Wenn sich ein PWM-Befehl in der Ausführungsphase des Ausgangsprofils befindet.
- Gelöscht (0) - Wenn sich ein PWM-Befehl nicht in der Ausführungsphase des Ausgangsprofils befindet.

## PWM-Beschleunigungsstatus (AS)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| AS - Beschleunigungs-status | PWM:0/AS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PWM-Bit AS (Beschleunigungsstatus) wird durch das PWM-Sub- system gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden. Funktionsweise des AS-Bits:

- Gesetzt (1) - Wenn sich ein PWM-Ausgang in der Beschleunigungsphase des Ausgangsprofils befindet.
- Gelöscht (0) - Wenn sich ein PWM-Ausgang nicht in der Beschleunigungsphase des Ausgangsprofils befindet.

## PWM-Profilparameterauswahl (PP)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| PP - Profilparameter-auswahl | PWM:0/PP | Bit | 0 oder 1 | Steuerung | Lesen/ Schreiben |

Die PWM PP (Profilparameterauswahl) wählt aus, welche Kompo- nente der Signalform während einer Flankenphase geändert wird:

- Gesetzt (1) - Frequenz auswählen
- Gelöscht (0) - Tastgrad auswählen

Das PWM-Bit PP kann nicht geändert werden, solange der PWM- Ausgang aktiv ist. Weitere Informationen finden Sie unter der Beschreibung von „PWM ADD“ auf Seite 6-26.

## PWM-Leerlaufstatus (IS)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| IS - PWM-Leerlauf-status | PWM:0/IS | Bit | 0 oder 1 | Status | Nur Lesen |

Der PWM-Bit IS (Leerlaufstatus) wird durch das PWM-Subsystem gesteuert und zeigt an, dass keine PWM-Aktivität vorliegt. Es kann von Eingangsbefehlen in dem Steuerprogramm verwendet werden.

- Gesetzt (1) - Das PWM-Subsystem befindet sich in einem Leerlaufstatus.
- Gelöscht (0) - Das PWM-Subsystem befindet sich nicht in einem Leerlaufstatus (es wird ausgeführt).

## PWM-Fehler erkannt (ED)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| ED - PWM-Fehler-erkennung | PWM:0/ED | Bit | 0 oder 1 | Status | Nur Lesen |

Das PWM-Bit ED (Fehler erkannt) wird durch das PWM-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden, um festzustellen, ob sich der PWM-Befehl in einem Fehlerzustand befindet. Wenn ein Fehlerzustand erkannt wird, wird der betreffende Fehler in dem Fehlercoderegister (PWM:0.ED) gekennzeichnet.

- Gesetzt (1) - Wenn sich ein PWM-Befehl in einem Fehlerzustand befindet.
- Gelöscht (0) - Wenn sich ein PWM-Befehl nicht in einem Fehlerzustand befindet.

## PWM-Normalbetrieb (NS)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| NS - PWM-Normalbetrieb | PWM:0/NS | Bit | 0 oder 1 | Status | Nur Lesen |

Das PWM-Bit NS (Normalbetrieb) wird durch das PWM-Subsystem gesteuert. Es kann von Eingangsbefehlen in jedem Strompfad innerhalb des Steuerprogramms verwendet werden, um festzustellen, ob sich der PTO-Befehl in seinem normalen Betriebsstatus befindet. Zu den normalen Betriebsstatus gehören: ACCEL (Beschleunigung), RUN (Ausführung) und DECEL (Verzögerung), sofern dabei keine PWM-Fehler vorliegen.

- Gesetzt (1) - Wenn sich ein PWM-Befehl in seinem Normalzustand befindet.
- Gelöscht (0) - Wenn sich ein PWM-Befehl nicht in seinem Normalzustand befindet.

## PWM-Hard-Stop aktivieren (EH)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| EH - PWM Hardstopp aktivieren | PWM:0/EH | Bit | 0 oder 1 | Steuerung | Lesen/Schreiben |

Mit dem PWM-Bit EH (Hardstop aktivieren) kann das PWM-Subsystem sofort angehalten werden. Bei einem PWM-Hardstopp wird ein PWM-Subsystemfehler erzeugt.

- Gesetzt (1) - Das PWM-Subsystem wird angewiesen, die Ausgangsmodulation sofort anzuhalten (Ausgang aus = 0).
- Gelöscht (0) - Normalbetrieb

## PWM-Status aktiv (ES)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| ES - PWM Status Aktiv | PWM:0/ES | Bit | 0 oder 1 | Status | Nur Lesen |

Das PWM-Bit ES (Status Aktiv) wird durch das PWM-Subsystem gesteuert. Wenn der Strompfad vor dem PWM-Befehl wahr ist, wird der PWM-Befehl aktiviert und das Status-Bit Aktiv gesetzt. Wenn der Strompfad vor dem PWM-Befehl in einen unwahren Zustand übergeht, wird das Status-Bit Aktiv sofort rückgesetzt (0).

- Gesetzt (1) - PWM aktiv
- Gelöscht (0) - PWM abgeschlossen oder Strompfad vor PWM ist unwahr

## PWM-Ausgangsfrequenz (OF)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| OF - PWM-Ausgangs-frequenz | PWM:0.OF | Wort (INT) | 0 bis 20000 | Steuerung | Lesen/Schreiben |

Die PWM-Variable OF (Ausgangsfrequenz) definiert die Frequenz des PWM-Ausgangs. Diese Frequenz kann jederzeit geändert werden.

## PWM-Betriebsfrequenzstatus (OFS)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| OFS - PWM-Betriebs-frequenzstatus | PWM:0.OFS | Wort (INT) | 0 bis 20000 | Status | Nur Lesen |

Der PWM-OFS (Betriebsfrequenzstatus) wird durch das PWM-Sub- system generiert und kann in dem Steuerprogramm zur Überwachung der tatsächlichen Frequenz verwendet werden, die durch das PWM-Subsystem erzeugt wird.

## PWM-Tastgrad (DC)

| Beschreibung Element | Adresse | Datenformat | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| DC - PWM-Tastgrad | PWM:0.DC | Wort (INT) | 1 bis 1000 | Steuerung | Lesen/ Schreiben |

Die PWM-Variable DC (Tastgrad) steuert das Ausgangssignal, das von dem PWM-Subsystem erzeugt wird. Bei Änderung dieser Variablen in dem Steuerprogramm ändert sich auch das Ausgangswellenprofil. Typische Werte und Ausgangswellenprofile:

- DC = 1000: 100 % Ausgang EIN (konstant, keine Welle)
- DC = 750: 75 % Ausgang EIN, 25 % Ausgang AUS
- DC = 500: 50 % Ausgang EIN, 50 % Ausgang AUS
- DC = 250: 25 % Ausgang EIN, 75 % Ausgang AUS
- DC = 0: 0 % Ausgang AUS (konstant, keine Welle)

## PWM-Tastgradstatus (DCS)

| Beschreibung Element | Adresse | Daten-format | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| DCS - PWM-Tastgrad-status | PWM:0.DCS | Wort (INT) | 1 bis 1000 | Status | Nur Lesen |

Die PWM-Variable DCS (Tastgradstatus) liefert die Rückführung von dem PWM-Subsystem. Die DCS-Variable kann innerhalb eines Eingangsbefehls auf einem Logikstrompfad verwendet werden, um dem übrigen Steuerprogramm den PWM-Systemstatus mitzuteilen.

## PWM-Beschleunigungs-/Bremsverzögerung (ADD)

| Beschreibung Element | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| ADD - Beschleunigungs-/ Bremsverzögerung | PWM:0.ADD | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/ Schreiben |

PWM ADD (Accel/Decel Delay) definiert die Zeitdauer in 10-Millisekunden-Intervallen, die für die Rampe von Null bis zur angegebenen Frequenz oder Dauer erforderlich sind. Zusätzlich wird die Zeit für die Rampe bis Null angegeben.

Der PWM ADD-Wert wird sofort geladen und aktiviert (immer dann, wenn der PWM-Befehl mit einer wahren Strompfadlogik abgetastet wird). Dies ermöglicht auch das Auftreten mehrerer Schritte oder Stufen der Beschleunigung/Verzögerung.

## PWM-Fehlercode (ER)

| Beschreibung Element | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| ER - PWM-Fehlercodes | PWM:0.ER | Wort (INT) | -2 bis 5 | Status | Nur Lesen |

PWM-Fehlercodes (ER), die von dem PWM-Subsystem erkannt wurden, werden in diesem Register angezeigt. Die bekannten Fehler sind in der nachfolgenden Tabelle aufgeführt:

| Fehlercode | Nicht vom Anwender verursachter Fehler | Korrigierbarer Fehler | Befehlsfehler | Fehlername | Beschreibung |
|---|---|---|---|---|---|
| -2 | Ja | Nein | Nein | Überlappungsfehler | Eine Ausgangsüberlappung wurde festgestellt. Mehrere Funktionen wurden demselben physischen Ausgang zugewiesen. Dies ist ein Konfigurationsfehler. Ein Steuerungsfehler wird erzeugt, und die Anwenderfehlerroutine wird nicht ausgeführt. Beispiel: PWM0 und PWM1 versuchen, denselben Ausgang zu verwenden. |
| -1 | Ja | Nein | Nein | Ausgangsfehler | Ein ungültiger Ausgang wurde angegeben. Nur Ausgang 2 und Ausgang 3 stehen zur Verfügung. Dies ist ein Konfigurationsfehler. Ein Steuerungsfehler wird erzeugt, und die Anwenderfehlerroutine wird nicht ausgeführt. |
| 0 | | | | Normal | Normal (0 = kein Fehler aufgetreten) |
| 1 | Nein | Nein | Ja | Hardstopp-Fehler | Dieser Fehler wird erzeugt, wenn ein Hardstopp festgestellt wurde. Bei diesem Fehler wird kein Steuerungsfehler erzeugt. Dieser Fehler wird automatisch gelöscht, sobald die Hardstopp-Bedingung aufgehoben wird. |
| 2 | Nein | Nein | Ja | Ausgang forciert | Der konfigurierte PWM-Ausgang (2 oder 3) wird erzwungen. Diese Bedinung *muss* aufgehoben werden, da sonst der PWM-Befehl nicht ausgeführt werden kann. Bei diesem Fehler wird kein Steuerungsfehler erzeugt. Dieser Fehler wird automatisch gelöscht, sobald die forcierte Bedingung aufgehoben wird. |
| 3 | Ja | Ja | Nein | Frequenzfehler | Der Wert der Frequenz ist kleiner als 0 oder größer als 20 000. Bei diesem Fehler wird ein Steuerungsfehler erzeugt. Dieser Fehler kann durch die Logik innerhalb der Anwenderfehlerroutine gelöscht werden. |
| 4 | Reserviert | | | | |
| 5 | Ja | Ja | Nein | Tastgradfehler | Der PWM-Tastgrad ist kleiner als Null oder größer als 1000.<br>Bei diesem Fehler wird ein Steuerungsfehler erzeugt. Dieser Fehler kann durch die Logik innerhalb der Anwenderfehlerroutine gelöscht werden. |

# Relaisbefehle (Bitbefehle)

Verwenden Sie Relaisbefehle (Bitbefehle) zur Überwachung und/oder Steuerung von Bits in einem Daten- oder Funktionsfile, z. B. als Eingangs-Bits oder als Zeitwerksteuerung-Wort-Bits. Folgende Befehle werden in diesem Kapitel beschrieben:

| Befehl | Zweck | Seite |
|---|---|---|
| XIC - Auf geschlossen prüfen | Prüfen eines Bits auf einen EIN-Zustand | 7-1 |
| XIO - Auf offen prüfen | Prüfen eines Bits auf einen AUS-Zustand | 7-1 |
| OTE - Ausgang einschalten | EIN- oder AUSSCHALTEN eines Bits (nicht remanent) | 7-3 |
| OTL - Ausgang verriegeln | Dauerhaftes Setzen eines Bits auf EIN (remanent) | 7-4 |
| OTU - Ausgang entriegeln | Setzen eines Bits auf AUS (remanent) | 7-4 |
| ONS - Einzelimpuls | Übergang von AUS nach EIN erkennen | 7-5 |
| OSR - Steigender Einzelimpuls | Übergang von AUS nach EIN erkennen | 7-6 |
| OSF - Fallender Einzelimpuls | Übergang von EIN nach AUS erkennen | 7-6 |

Diese Befehle arbeiten mit einem Daten-Bit. Während des Betriebs kann der Prozessor je nach der logischen Kontinuität der Strompfade das Bit setzen bzw. rücksetzen. Sie können ein Bit so oft adressieren, wie es das Programm erfordert.

## XIC – Auf geschlossen prüfen
## XIO – Auf offen prüfen

Befehlstyp: Eingang

**Tabelle 7.1 Ausführungszeit für die Befehle XIC und XIO**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 0,9 µs | 0,8 µs |
| MicroLogix 1500 | 0,9 µs | 0,7 µs |

Verwenden Sie den Befehl XIC, um festzustellen, ob das adressierte Bit gesetzt ist. Verwenden Sie den Befehl XIO, um festzustellen, ob das adressierte Bit rückgesetzt ist.

Bei Verwendung auf einem Strompfad kann die zu prüfende Bitadresse dem Status realer Eingangsgeräte, die an die Basiseinheit oder Erweiterungs-E/A

angeschlossen sind, oder internen Adressen (Daten- oder Funktionsfiles) entsprechen. Folgende Geräte können z. B. ein- bzw. ausgeschaltet werden:

- ein Drucktaster, der mit einem Eingang verdrahtet ist (als I1:0/4 adressiert)
- ein Ausgang, der mit einer Kontrolllampe verdrahtet ist (als O0:0/2 adressiert)
- ein Zeitwerk, das eine Lampe steuert (als T4:3/DN adressiert)
- ein Bit in einem Bitfile (als B3/16 adressiert)

Funktionsweise der Befehle:

**Tabelle 7.2 Funktionsweise der Befehle XIO und XIC**

| Strompfad-status | Adressiertes Bit | XIC-Befehl | XIO-Befehl |
|---|---|---|---|
| wahr | Aus | Erzeugt Antwort unwahr | Erzeugt Antwort wahr |
| wahr | Ein | Erzeugt Antwort wahr | Erzeugt Antwort unwahr |
| unwahr | -- | Befehl wird nicht geprüft | Befehl wird nicht geprüft |

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 7.3 Gültige Adressierungsmodi und Filetypen für XIC- und XIO-Befehle**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles(1) | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll (2) | Adressierungsmodus(3) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Operanden-Bit | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | • | | | |

(1) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur zur Verwendung mit MicroLogix 1200- und 1500 BXB-Geräten eingesetzt.

(2) Der Datenprotokollierungs-Statusfile kann nur durch den MicroLogix 1500 1764-LRP-Prozessor verwendet werden.

(3) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# OTE – Ausgang einschalten

Befehlstyp: Ausgang

**Tabelle 7.4 Ausführungszeit für OTE-Befehl**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 1,4 µs | 1,1 µs |
| MicroLogix 1500 | 1,2 µs | 0,0 µs |

Mit einem OTE-Befehl können Sie einen Bit-Standort einschalten, wenn der Strompfadstatus als wahr erkannt wird, und ausschalten, wenn der Strompfadstatus als unwahr erkannt wird. Ein Ausgang, der mit einer Kontrolllampe verdrahtet ist (als O0:0/4 adressiert), ist z. B. ein Gerät, das ein- bzw. ausgeschaltet werden kann. OTE-Befehle werden rückgesetzt (auf AUS gesetzt), wenn:

- Sie den Programm-Modus oder den fernen Programm-Modus (wieder) aktivieren oder die Spannungsversorgung wieder herstellen.
- der OTE-Befehl innerhalb einer inaktiven oder unwahren MCR-Zone (Master Control Reset) programmiert wird.

**HINWEIS** Ein Bit, das innerhalb eines Unterprogramms mit einem OTE-Befehl gesetzt wird, bleibt gesetzt, bis der OTE-Befehl erneut abgefragt wird.

**ACHTUNG**

Wenn Interrupts während der Programmabfrage über einen OTL-, OTE- oder UIE-Befehl aktiviert werden, *muss* dieser Befehl der *letzte* Befehl sein, der auf dem Strompfad ausgeführt wird (letzter Befehl auf letztem Abzweig). Es wird empfohlen, in dem Strompfad diesen Befehl als einzigen Ausgangsbefehl zu verwenden.

**ACHTUNG**

Verwenden Sie eine Ausgangsadresse immer nur an einer Stelle innerhalb des Logikprogramms. Bedenken Sie immer die Last, die die Ausgangsspule darstellt.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 7.5 Gültige Adressierungsmodi und Filetypen für den OTE-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles(1) | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll(2) | Adressierungsmodus(3) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Ziel-Bit | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | | | • | | | | • | | • | • | • | | | |

(1) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur zur Verwendung mit MicroLogix 1200- und 1500 BXB-Geräten eingesetzt

(2) Der Datenprotokollierungs-Statusfile kann nur durch den MicroLogix 1500 1764-LRP-Prozessor verwendet werden.

(3) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# OTL – Ausgang verriegeln
# OTU – Ausgang entriegeln

Befehlstyp: Ausgang

**Tabelle 7.6 Ausführungszeit für die Befehle OTL und OTU**

| Steuerung | OTL - Strompfad ist: | | OTU - Strompfad ist: | |
|---|---|---|---|---|
| | wahr | unwahr | wahr | unwahr |
| MicroLogix 1200 | 1,0 µs | 0,0 µs | 1,1 µs | 0,0 µs |
| MicroLogix 1500 | 0,9 µs | 0,0 µs | 0,9 µs | 0,0 µs |

Die Befehle OTL und OTU sind speichernde Ausgangsbefehle. Mit OTL wird ein Bit gesetzt, während mit OTU ein Bit rückgesetzt wird. Diese Befehle werden normalerweise paarweise verwendet, wobei beide Befehle dasselbe Bit adressieren.

**ACHTUNG**

Wenn Interrupts während der Programmabfrage über einen OTL-, OTE- oder UIE-Befehl aktiviert werden, *muss* dieser Befehl der *letzte* Befehl sein, der auf dem Strompfad ausgeführt wird (letzter Befehl auf letztem Abzweig). Es wird empfohlen, in dem Strompfad diesen Befehl als einzigen Ausgangsbefehl zu verwenden.

Ein Ausgang, der auf diese Weise gesetzt oder rückgesetzt wurde, behält diesen Status unabhängig von dem Strompfadstatus bei.

**ACHTUNG**

Im Falle eines Spannungsausfalls werden alle mit einem OTL-Befehl gesteuerten Bits (einschließlich Feldgeräte) beim Wiederherstellen der Spannung eingeschaltet, wenn das OTL-Bit zum Zeitpunkt des Spannungsausfalls gesetzt war.

**ACHTUNG**

Kommt es zu einem nicht behebbaren Fehler, werden die physischen Ausgänge ausgeschaltet. Nachdem der Fehler gelöscht wurde, wird der Steuerungsbetrieb entsprechend des Datentafelwerts fortgesetzt.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 7.7 Gültige Adressierungsmodi und Filetypen für OTL- und OTU-Befehle**
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles(1) | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll (2) | Adressierungsmodus(3) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Operanden-Bit | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | | | • | | | | • | | • | • | • | | | |

(1) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur zur Verwendung mit MicroLogix 1200- und 1500 BXB-Geräten eingesetzt.

(2) Der Datenprotokollierungs-Statusfile kann nur durch den MicroLogix 1500 1764-LRP-Prozessor verwendet werden.

(3) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

## ONS – Einzelimpuls

N7:1
ONS
0

Befehlstyp: Eingang

**Tabelle 7.8 Ausführungszeit für ONS-Befehl**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 2,6 µs | 1,9 µs |
| MicroLogix 1500 | 2,2 µs | 1,7 µs |

**HINWEIS** Der ONS-Befehl für die MicroLogix 1200- und 1500-Steuerungen bietet dieselbe Funktionalität wie der OSR-Befehl für die MicroLogix 1000- und SLC 500-Steuerungen.

Der ONS-Befehl ist ein speichernder Eingangsbefehl, der ein einmaliges Ereignis auslöst. Nach Übergang des Strompfadzustands von unwahr zu wahr bleibt der ONS-Befehl während einer Programm- abfrage wahr. Dann wird der Ausgang auf AUS gesetzt, bis die Logik vor dem ONS-Befehl unwahr wird; (danach wird der ONS-Befehl wieder aktiviert).

Der Strompfadzustand der vorherigen Abfrage wird in dem ONS-Speicher-Bit festgehalten. Dieses Bit wird zur Erinnerung an den Wechsel des Strompfadstatus von unwahr nach wahr verwendet.

**Tabelle 7.9 Funktionsweise des ONS-Befehls:**

| Strompfadübergang | Speicher-Bit | Strompfadzustand nach Ausführung |
|---|---|---|
| unwahr zu wahr (eine Abfrage) | Speicher-Bit gesetzt | wahr |
| wahr zu wahr | Speicher-Bit bleibt gesetzt | unwahr |
| wahr zu unwahr, unwahr zu unwahr | Speicher-Bit wird rückgesetzt | unwahr |

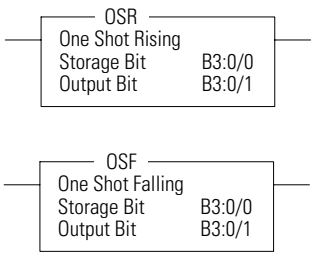
Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 7.10 Gültige Adressierungsmodi und Filetypen für ONS-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Speicher-Bit | | | | • | | • | | | | | | | | | | | | | | | | | | | • | | • | | | |

# OSR – Steigender Einzelimpuls
# OSF – Abfallender Einzelimpuls




Befehlstyp: Ausgang

**Tabelle 7.11 Ausführungszeit für die Befehle OSR und OSF**

| Steuerung | OSR - Strompfad ist: | | OSF - Strompfad ist: | |
|---|---|---|---|---|
| | wahr | unwahr | wahr | unwahr |
| MicroLogix 1200 | 3,4 µs | 3,0 µs | 2,8 µs | 3,7 µs |
| MicroLogix 1500 | 3,2 µs | 2,8 µs | 2,7 µs | 3,4 µs |

**HINWEIS** Der OSR-Befehl für die MicroLogix 1200- und 1500-Steuerungen bietet *nicht* dieselbe Funktionalität wie der OSR-Befehl für die MicroLogix 1000- und SLC 500-Steuerungen. Wenn Sie dieselbe Funktio- nalität wünschen, die der OSR-Befehl für die MicroLogix 1000- und SLC 500-Steuerungen bietet, verwenden Sie den ONS-Befehl.

Mit den Befehlen OSR und OSF wird ein einmaliges Ereignis ausgelöst. Diese Auslösung hängt ab von der Änderung des Strompfadzustands:

- Mit dem OSR-Befehl wird ein Ereignis bei einem Strompfad- übergang von unwahr nach wahr (Anstiegsflanke) ausgelöst.
- Mit dem OSF-Befehl wird ein Ereignis bei einem Strompfad- übergang von wahr nach unwahr (abfallende Flanke) ausgelöst.

Diese Befehle verwenden zwei Parameter, das Speicher-Bit und das Ausgangs-Bit.

- Speicher-Bit - Unter dieser Bitadresse wird der Strompfadzustand nach der vorherigen Abfrage gespeichert.
- Ausgangs-Bit - Dieses Bit wird bei einem Strompfadübergang von unwahr nach wahr (OSR) oder wahr nach unwahr (OSF) gesetzt. Das Ausgangs-Bit wird für eine Programmabfrage gesetzt.

Um den OSR-Befehl erneut zu aktivieren, muss der Strompfad unwahr werden. Um den OSR-Befehl erneut zu aktivieren, muss der Strom- pfad wahr werden.

**Tabelle 7.12 Funktionsweise des Speicher-Bits und des Ausgangs-Bits bei OSR**

| Strompfadübergang | Speicher-Bit | Ausgangs-Bit |
|---|---|---|
| unwahr zu wahr (eine Abfrage) | Bit wird gesetzt | Bit wird gesetzt |
| wahr zu wahr | Bit wird gesetzt | Bit wird gesetzt |
| wahr zu unwahr, unwahr zu unwahr | Bit wird gesetzt | Bit wird gesetzt |

**Tabelle 7.13 Funktionsweise des Speicher-Bits und des Ausgangs-Bits bei OSF**

| Strompfadübergang | Speicher-Bit | Ausgangs-Bit |
|---|---|---|
| wahr zu unwahr (eine Abfrage) | Bit wird gesetzt | Bit wird gesetzt |
| unwahr zu unwahr | Bit wird gesetzt | Bit wird gesetzt |
| unwahr zu wahr, wahr zu wahr | Bit wird gesetzt | Bit wird gesetzt |

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 7.14 Gültige Adressierungsmodi und Filetypen für die OSR- und OSF-Befehle**

*Definitionen der in dieser Tabelle verwendeten Begriffen finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Speicher-Bit | | | | • | | • | | | | | | | | | | | | | | | | | | | • | | • | | | |
| Ausgangs-Bit | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | | • | | • | | | |

# Zeitwerk- und Zählerbefehle

Die Zeitwerk- und Zählerbefehle sind Ausgangsbefehle, die zeitabhängige Vorgänge oder Vorgänge, die von der Anzahl von Ereignissen abhängen, steuern. In diesem Kapitel werden folgende Zeitwerk- und Zählerbefehle beschrieben:

| Befehl | Zweck | Seite |
|---|---|---|
| TON - Timer-Einschaltverzögerung | Verzögerung des Einschaltens eines Ausgangs auf einem wahren Strompfad | 8-4 |
| TOF - Timer-Ausschaltverzögerung | Verzögerung des Ausschaltens eines Ausgangs auf einem unwahren Strompfad. | 8-5 |
| RTO - Speichernder Timer Ein | Verzögerung des Einschaltens eines Ausgangs von einem wahren Strompfad. Der Istwert wird gespeichert. | 8-6 |
| CTU - Aufwärtszählung | Aufwärtszählung | 8-9 |
| CTD - Abwärtszählung | Abwärtszählung | 8-9 |
| RES - Rücksetzen | Zurücksetzen des RTO sowie des Istwerts und der Status-Bits eines Zählers (wird bei TOF-Timern nicht verwendet). | 8-10 |

Informationen zur Verwendung der Ausgänge für Hochgeschwindigkeitszähler finden Sie im Abschnitt „Verwenden des Hochgeschwindigkeitszählers und des programmierbaren Endschalters“ auf Seite 5-1.

## Zeitwerkbefehle – Übersicht

In einer Steuerung befinden sich die Zeitwerke in einem Zeitwerkfile. Ein beliebiger, nicht benutzter Datenfile kann als Zeitwerkfile verwendet werden. Wenn ein Datenfile als Zeitwerkfile verwendet wird, enthält jedes Element innerhalb des Zeitwerkfiles drei Unterelemente:

- Zeitwerksteuerung und -status
- Sollwert - Bei Erreichen dieses Werts durch das Zeitwerk wird das Zeitwerk ausgeschaltet. Wenn der Istwert diesen Wert erreicht, wird das DN-Status-Bit (Fertig) gesetzt (nur bei TON und RTO). Der Datenbereich des Sollwerts ist von 0 bis 32767. Das kleinste Aktualisierungsintervall beträgt unabhängig von der Zeitbasis 2,55 Sekunden.
- Istwert - Der Istwert zählt die Zeitbasisintervalle. Damit gibt der Istwert die verstrichene Zeit wider. Der Datenbereich des Istwerts ist von 0 bis 32767.

Für Zeitwerke stehen drei Zeitbasen zur Verfügung:

**Tabelle 8.1 Zeitbasiseinstellungen**

| Zeitbasis | Bereich der Zeitmessung |
|---|---|
| 0,001 Sekunden | 0 bis 32,767 Sekunden |
| 0,01 Sekunden | 0 bis 327,67 Sekunden |
| 1,00 Sekunden | 0 bis 32,767 Sekunden |

Jede Zeitwerkadresse besteht aus einem 3 Worte umfassenden Element. Wort 0 ist das Steuer- und Statuswort, in Wort 1 wird der Sollwert und in Wort 2 der Istwert gespeichert.

**Tabelle 8.2 Zeitwerkfile**

| Wort | Bit | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Wort 0 | EN | TT | DN | zur internen Verwendung | | | | | | | | | | | | |
| Wort 1 | Sollwert | | | | | | | | | | | | | | | |
| Wort 2 | Istwert | | | | | | | | | | | | | | | |

EN = Freigabe-Bit Zeitwerk

TT = Zeitwerk-aktiv-Bit

DN = Fertig-Bit

**ACHTUNG**

Kopieren Sie die Zeitwerkelemente nicht, solange das Freigabe-Bit für das Zeitwerk (EN) gesetzt ist. Dies könnte zu einem unvorhergesehenen Verhalten der Maschine führen.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 8.3 Gültige Adressierungsmodi und Filetypen für Zeitwerkbefehle**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles(1) | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Zeitwerk | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Zeitbasis | | | | | | | | | | | | | | | | | | | | | | | | • | | | | | | • |
| Sollwert | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Istwert | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

(1) Nur gültig für Zeitwerkfiles.

**HINWEIS** Den Istwert und die Status-Bits eines Zeitwerks können Sie mit dem Befehl RES rücksetzen.

## Zeitwerkgenauigkeit

Die Zeitwerkgenauigkeit bezieht sich auf die Zeitspanne zwischen der Aktivierung eines Zeitwerkbefehls und der Beendigung des gemessenen Zeitintervalls.

**Tabelle 8.4 Zeitwerkgenauigkeit**

| Zeitbasis | Genauigkeit |
|---|---|
| 0,001 Sekunden | -0,001 bis 0,00 |
| 0,01 Sekunden | -0,01 bis 0,00 |
| 1,00 Sekunden | -1,00 bis 0,00 |

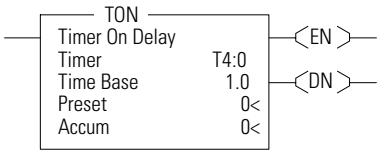
Wenn Ihre Programmabfrage 2,5 Sekunden überschreiten kann, wiederholen Sie den Zeitwerkbefehl auf einem anderen Strompfad (identische Logik) und in einem anderen Bereich des Kontakt- plancodes, so dass der Strompfad innerhalb dieser Grenzen abgefragt wird.

## Wiederholung von Zeitwerkbefehlen

Das Freigabe-Bit (EN) eines Zeitwerks bietet eine einfache Möglichkeit zur Wiederholung der komplexen Bedingungslogik des Zeitwerks auf einem anderen Strompfad innerhalb des Kontaktplanprogramms.

**HINWEIS** Die Zeitmessung ist möglicherweise ungenau, wenn der Strompfad, in dem sich der aktive Zeitwerkbefehl befindet, durch Befehle wie JMP (Sprung zu Marke), LBL (Label), JSR (Sprung ins Unterprogramm) oder SBR (Unterprogramm) übersprungen wird. Wenn dieser Sprung innerhalb von 2,5 Sekunden erfolgt, tritt kein Zeitverlust auf. Bei Sprüngen von mehr als 2,5 Sekunden tritt ein unentdeckter Zeitmessfehler auf. Bei Unterprogrammen muss ein Zeitwerk mindestens alle 2,5 Sekunden abgefragt werden, um Zeitmessfehler zu vermeiden.

# TON – Timer-Einschaltverzögerung




Befehlstyp: Ausgang

**Tabelle 8.5 Ausführungszeit für TON-Befehl**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 18,0 µs | 3,0 µs |
| MicroLogix 1500 | 15,5 µs | 2,5 µs |

Verwenden Sie den TON-Befehl, um das Einschalten eines Ausgangs zu verzögern. Der TON-Befehl beginnt mit dem Zählen von Zeitbasisintervallen, sobald ein Strompfadstatus wahr wird. Solange die Strompfadbedingung wahr ist, erhöht sich der Istwert des Zeitwerks bis zum Erreichen des Sollwerts. Sobald der Istwert den Sollwert erreicht hat, wird die Zeitmessung angehalten.

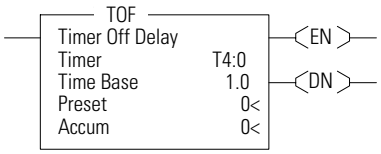
Unabhängig von Ist- und Sollwert wird der Istwert rückgesetzt (0), wenn der Strompfadstatus unwahr wird. Bei Aus- und Einschalten der Spannungsversorgung sowie bei Moduswechseln wird das TON-Zeitwerk rückgesetzt.

Zeitwerkbefehle verwenden folgende Steuer- und Status-Bits:

**Tabelle 8.6 Zeitwerksteuer- und -status-Bits, Zeitwerk-Wort 0 (Bei diesem Beispiel wurde der Datenfile 4 als Zeitwerkfile konfiguriert.)**

| Bit | | Gesetzt, wenn: | Bleibt gesetzt, bis: |
|---|---|---|---|
| **Bit 13 - T4:0/DN** | **DN - Fertig-Bit** | Istwert ≥ Sollwert | Strompfadstatus unwahr wird |
| **Bit 14 - T4:0/DN** | **TT - Zeitwerk aktiv** | Strompfadstatus unwahr und Istwert < Sollwert | • Strompfadstatus unwahr wird<br>• Fertig-Bit gesetzt wird |
| **Bit 15 - T4:0/EN** | **EN - Freigabe-Bit** | Strompfadstatus wahr | Strompfadstatus unwahr wird |

# TOF – Timer-Ausschaltverzögerung




Befehlstyp: Ausgang

**Tabelle 8.7 Ausführungszeit für TOF-Befehl**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 2,9 µs | 13,0 µs |
| MicroLogix 1500 | 2,5 µs | 10,9 µs |

Verwenden Sie den TOF-Befehl, um das Ausschalten eines Ausgangs zu verzögern. Der TOF-Befehl beginnt mit dem Zählen von Zeitbasisintervallen, sobald ein Strompfadstatus unwahr wird. Solange die Strompfadbedingung unwahr ist, erhöht sich der Istwert des Zeitwerks bis zum Erreichen des Sollwerts.

Unabhängig von Ist- und Sollwert wird der Istwert rückgesetzt (0), wenn der Strompfadstatus wahr wird. Bei Aus- und Einschalten der Spannungsversorgung sowie bei Moduswechseln wird das TOF- Zeitwerk rückgesetzt.

Zeitwerkbefehle verwenden folgende Steuer- und Status-Bits:

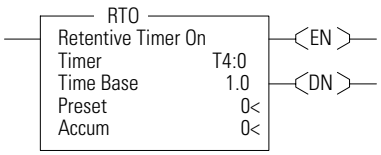
**Tabelle 8.8 Zeitwerksteuer- und -status-Bits, Zeitwerk-Wort 0 (Bei diesem Beispiel wurde der Datenfile 4 als Zeitwerkfile konfiguriert.)**

| Bit | | Gesetzt, wenn: | Bleibt gesetzt, bis: |
|---|---|---|---|
| **Bit 13 - T4:0/DN** | **DN - Fertig-Bit** | die Strompfadbedingungen wahr sind | die Strompfadbedingungen unwahr werden und der Istwert größer als oder gleich dem Sollwert ist |
| **Bit 14 - T4:0/DN** | **TT - Zeitwerk aktiv** | die Strompfadbedingungen unwahr sind und der Istwert kleiner als der Sollwert ist | die Strompfadbedingungen wahr werden oder das Fertig-Bit rückgesetzt wird |
| **Bit 15 - T4:0/EN** | **EN - Freigabe-Bit** | die Strompfadbedingungen wahr sind | die Strompfadbedingungen unwahr werden |

**ACHTUNG**

Da der Istwert und die Status-Bits durch den RES-Befehl rückgesetzt werden, sollten Sie den RES-Befehl nicht verwenden, um eine Zeitwerk- adresse in einem TOF-Befehl rückzusetzen. Wenn der Istwert und die Status-Bits des TOF-Zeitwerks rückgesetzt werden, kann die Maschine außer Kontrolle geraten.

# RTO – Speichernder Timer Ein




Befehlstyp: Ausgang

**Tabelle 8.9 Ausführungszeit für RTO-Befehl**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 18,0 µs | 2,4 µs |
| MicroLogix 1500 | 15,8 µs | 2,2 µs |

Verwenden Sie den RTO-Befehl, um das Einschalten eines Ausgangs zu verzögern. Der RTO-Befehl beginnt mit dem Zählen von Zeitbasisintervallen, sobald ein Strompfadstatus wahr wird. Solange die Strompfadbedingung wahr ist, erhöht sich der Istwert des Zeitwerks bis zum Erreichen des Sollwerts.

Der Istwert des RTO-Befehls bleibt erhalten, wenn eines der folgenden Ereignisse eintritt:

- die Strompfadbedingungen werden wahr
- Sie schalten die Steuerung von dem Run oder Testmodus in den Programm-Modus um
- die Spannungsversorgung des Prozessors wird unterbrochen
- ein Fehler tritt auf.

Wenn Sie wieder in den Run- oder Testmodus zurückwechseln und/oder der Strompfadstatus wahr wird, wird die Zählung ab dem gespeicherten Istwert fortgesetzt. Bei Aus- und Einschalten der Spannungsversorgung sowie bei Moduswechseln wird das RTO-Zeitwerk nicht rückgesetzt.

Zeitwerkbefehle verwenden folgende Steuer- und Status-Bits:

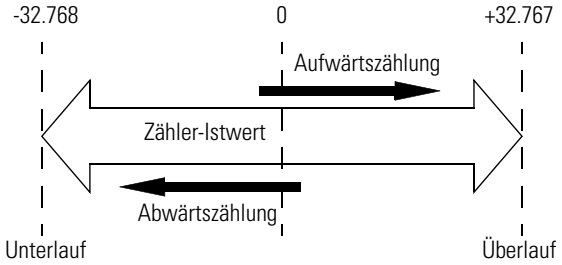
**Tabelle 8.10 Zählersteuer- und -status-Bits, Zeitwerk-Wort 0 (Bei diesem Beispiel wurde der Datenfile 4 als Zeitwerkfile konfiguriert.)**

| Bit | | Gesetzt, wenn: | Bleibt gesetzt, bis: |
|---|---|---|---|
| **Bit 13 - T4:0/DN** | **DN - Fertig-Bit** | Istwert ≥ Sollwert | der entsprechende RESBefehl aktiviert wird |
| **Bit 14 - T4:0/DN** | **TT - Zeitwerk aktiv** | Strompfadstatus unwahr und Istwert < Sollwert | • Strompfadstatus unwahr wird oder<br>• Fertig-Bit gesetzt wird |
| **Bit 15 - T4:0/EN** | **EN - Freigabe-Bit** | Strompfadstatus wahr | Strompfadstatus unwahr wird |

Den Istwert eines speichernden Zeitwerks können Sie mit dem Befehl RES rücksetzen. Siehe „RES – Zurücksetzen“ auf Seite 8-10.

## Funktionsweise von Zählern

Die folgende Abbildung veranschaulicht die Funktionsweise von Zählern. Der Zählwert muss im Bereich von –32 768 bis +32 767 liegen. Wenn der Zählwert den Grenzwert von +32 767 überschreitet, wird das Überlauf-Bit für den Zählerstatus (OV) gesetzt (1). Wenn der Zählwert den Grenzwert von –32 768 unterschreitet, wird das Unterlauf-Bit für den Zählerstatus (UN) gesetzt (1). Der Zähler kann mit dem RES-Befehl (Rücksetzen) rückgesetzt (0) werden.




### CTU- und CTD-Befehle verwenden

Zählerbefehle werden mit den folgenden Parametern verwendet:

- Zähler - Dies ist die Adresse des Zählers innerhalb des Datenfiles. Alle Zähler sind 3-Wort-Datenelemente. Wort 0 enthält die Steuer- und Status-Bits, Wort 1 den Sollwert und Wort 2 den Istwert.

| Wort | Bit | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Wort 0 | CU | CD | DN | OV | UN | nicht verwendet | | | | | | | | | | |
| Wort 1 | Sollwert | | | | | | | | | | | | | | | |
| Wort 2 | Istwert | | | | | | | | | | | | | | | |

CU = Aufwärtszählung-Freigabe-Bit
CD = Abwärtszählung-Freigabe-Bit
DN = Fertig-Bit
OV = Aufwärtszählung-Überlauf-Bit
UN = Abwärtszählung-Unterlauf-Bit

- Sollwert - Wenn der Istwert diesen Wert erreicht, wird das DN-Bit (Fertig) gesetzt. Der Datenbereich des Sollwerts ist von -32768 bis 32767.
- Istwert - Der Istwert enthält den aktuellen Zählwert. Der Datenbereich des Istwerts ist von -32768 bis 32767.

  Der Istwert wird bei jedem Strompfadübergang von unwahr nach wahr erhöht (CTU) oder reduziert (CTD). Der Istwert bleibt gespeichert, wenn der Strompfadstatus wieder unwahr oder die Steuerung aus- und wieder eingeschaltet wird. Der Istwert bleibt gespeichert, bis er durch einen RES-Befehl (Rücksetzen) gelöscht wird, der dieselbe Adresse wie der Zähler hat.

**HINWEIS** Wenn der Istwert den CTU-Sollwert überschreitet oder den CDT-Sollwert unterschreitet, wird die Zählung fortgesetzt.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 8.11 Gültige Adressierungsmodi und Filetypen für die CTD- und CTU-Befehle**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles(1) | | | | | | | | | | | Funktionsfiles | | | | | | | | | | | | | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | CS - Komm | IOS - E/A | DLS - Datenprotokoll | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Zähler | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Sollwert | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Istwert | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

(1) Nur gültig für Zählerfiles.

## Steuer- und Status-Bits des Zählerfiles verwenden

Wie der Istwert werden auch die Status-Bits des Zählers bis zum Rücksetzen gespeichert (siehe unten).

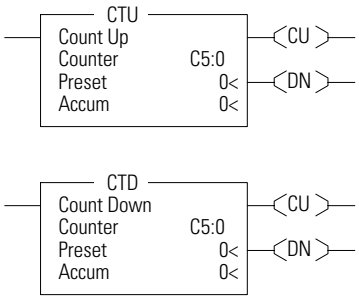
**Tabelle 8.12 Zählersteuer- und -status-Bits bei CTU-Befehl, Zähler-Wort 0 (Bei diesem Beispiel wurde der Datenfile 5 als Zählerfile konfiguriert.)**

| Bit | | Gesetzt, wenn: | Bleibt gesetzt, bis: |
|---|---|---|---|
| **Bit 12 - C5:0/OV** | **OV - Überlauf** | der Istwert wird von +32.767 auf -32.768 gesetzt, dann wird die Aufwärtszählung fortgesetzt | ein RES-Befehl mit derselben Adresse wie der CTU-Befehl wird ausgeführt |
| **Bit 13 - C5:0/DN** | **DN - Fertig** | Istwert ≥ Sollwert | • Istwert < Sollwert oder<br>• ein RES-Befehl mit derselben Adresse wie der CTU-Befehl wird ausgeführt |
| **Bit 15 - C5:0/CU** | **CU - Aufwärts-zählung** | Strompfadstatus wahr | • Strompfadstatus unwahr<br>• ein RES-Befehl mit derselben Adresse wie der CTU-Befehl wird ausgeführt |

**Tabelle 8.13 Zählersteuer- und status-Bits bei CTD-Befehl, Zähler-Wort 0 (Bei diesem Beispiel wurde der Datenfile 5 als Zählerfile konfiguriert.)**

| Bit | | Gesetzt, wenn: | Bleibt gesetzt, bis: |
|---|---|---|---|
| **Bit 11 - C5:0/UN** | **UN - Unterlauf** | der Istwert wird von -32.768 auf +32.767 gesetzt, dann wird die Abwärtszählung fortgesetzt | ein RES-Befehl mit derselben Adresse wie der CTD-Befehl wird ausgeführt |
| **Bit 13 - C5:0/DN** | **DN - Fertig** | Istwert ≥ Sollwert | • Istwert < Sollwert oder<br>• ein RES-Befehl mit derselben Adresse wie der CTU-Befehl wird ausgeführt |
| **Bit 14 - C5:0/CD** | **CD - Abwärts-zählung** | Strompfadstatus wahr | • Strompfadstatus unwahr<br>• ein RES-Befehl mit derselben Adresse wie der CTD-Befehl wird ausgeführt |

# CTU – Aufwärtszählung
# CTD – Abwärtszählung




Befehlstyp: Ausgang

**Tabelle 8.14 Ausführungszeit für die Befehle CTU und CTD**

| Steuerung | CTU - Strompfad ist: | | CTD - Strompfad ist: | |
|---|---|---|---|---|
| | wahr | unwahr | wahr | unwahr |
| MicroLogix 1200 | 9,0 µs | 9,2 µs | 9,0 µs | 9,0 µs |
| MicroLogix 1500 | 6,4 µs | 8,5 µs | 7,5 µs | 8,5 µs |

Mit den Befehlen CTU und CTD wird ein Zähler bei jedem Strompfadübergang von unwahr nach wahr erhöht oder reduziert. Bei einem CTU-Strompfadübergang von unwahr nach wahr wird der Istwert um ein Zählintervall erhöht. Bei einem CTD-Strompfadübergang von unwahr nach wahr wird der Istwert um ein Zählintervall reduziert.

**HINWEIS** Wenn das Signal von einem Feldgerät gesendet wird, das mit einem Ausgang der Steuerung verdrahtet ist, darf die Ein- und Aus-Dauer des eingehenden Signals maximal das Doppelte der Abfragezeit der Steuerung betragen (bei Annahme eines Tastgrades von 50 %). Diese Bedingung muss erfüllt sein, damit der Zähler in der Lage ist, Übergänge von unwahr nach wahr bei dem eingehenden Gerätesignal zu erkennen.

# RES – Zurücksetzen

Befehlstyp: Ausgang

**Tabelle 8.15 Ausführungszeit für RES-Befehl**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 5,9 µs | 0,0 µs |
| MicroLogix 1500 | 4,8 µs | 0,0 µs |

Mit dem RES-Befehl werden Zeitwerke, Zähler und Steuerelemente rückgesetzt. Bei Ausführung des RES-Befehls werden die in dem Befehl definierten Daten rückgesetzt.

Wenn der Strompfad unwahr ist, hat der RES-Befehl keinerlei Auswirkungen. Die folgende Tabelle zeigt, welche Elemente geändert werden:

**Tabelle 8.16 Funktionsweise des RES-Befehls:**

| Bei Verwendung eines RES-Befehls für ein: | | |
|---|---|---|
| **Zeitwerkelement** | **Zählerelement** | **Steuerelement** |
| setzt die Steuerung folgende Werte/Bits zurück: | setzt die Steuerung folgende Werte/Bits zurück: | setzt die Steuerung folgende Werte/Bits zurück: |
| Istwert auf 0 | Istwert auf 0 | POS-Wert auf 0 |
| DN-Bit | OV-Bit | EN-Bit |
| TT-Bit | UN-Bit | EU-Bit |
| EN-Bit | DN-Bit | DN-Bit |
| | CU-Bit | EM-Bit |
| | CD-Bit | ER-Bit |
| | | UL-Bit |

**ACHTUNG**

Da der Istwert und die Status-Bits durch den RES-Befehl rückgesetzt werden, sollten Sie den RES-Befehl nicht verwenden, um eine Zeitwerkadresse in einem TOF-Befehl rückzusetzen. Wenn der Istwert und die Status-Bits des TOF-Zeitwerks rückgesetzt werden, kann die Maschine außer Kontrolle geraten und Verletzungen verursachen.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 8.17 Gültige Adressierungsmodi und Filetypen für RES-Befehl**
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Struktur | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |

# Vergleichsbefehle

Verwenden Sie diese Eingangsbefehle, um Datenwerte zu vergleichen.

| Befehl | Zweck | Seite |
|---|---|---|
| EQU - Gleich | Prüfen, ob zwei Werte gleich sind (=) | 9-3 |
| NEQ - Ungleich | Prüfen, ob ein Wert ungleich einem zweiten Wert ist (≠) | 9-3 |
| LES - Kleiner als | Prüfen, ob ein Wert kleiner als ein zweiter Wert ist (<) | 9-4 |
| LEQ - Kleiner als oder gleich | Prüfen, ob ein Wert kleiner als oder gleich einem zweiten Wert ist (≤) | 9-5 |
| GRT - Größer als | Prüfen, ob ein Wert größer als ein zweiter Wert ist (>) | 9-4 |
| GEQ - Größer als oder gleich | Prüfen, ob ein Wert größer als oder gleich einem zweiten Wert ist (≥) | 9-5 |
| MEQ - Maskierter Vergleich auf gleich | Prüfen, ob Teile zweier Werte gleich sind | 9-6 |
| LIM - Grenzwerttest | Prüfen, ob ein Wert zwischen zwei anderen Werten liegt | 9-7 |

## Vergleichsbefehle verwenden

Bei den meisten Vergleichsbefehle werden zwei Parameter verwendet, Quelle A und Quelle B (ein zusätzlicher Parameter wird bei den Befehlen MEQ und LIM verwendet, siehe Beschreibung in diesem Kapitel). Bei beiden Quellen darf es sich nicht um unmittelbare Quellen handeln. Für diese Befehle gelten folgende Bereiche:

-32768 bis 32767 (Wort)
-2.147.483.648 bis 2.147.483.647 (Doppelwort)

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 9.1 Gültige Adressierungsmodi und Filetypen für die EQU-, NEQ-, GRT-, LES-, GEQ- und LEQ-Befehle**

*Definitionen der in dieser Tabelle verwendeten Begriffen finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles(1) | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll(2) | Adressierungsmodus(3) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F(4) | ST | L | MG, PD | PLS | RTC | HSC(5) | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle A | • | • | • | • | • | • | • | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | | • | • | |
| Quelle B | • | • | • | • | • | • | • | | • | • | | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |

(1) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur zur Verwendung mit MicroLogix 1200- und 1500 BXB-Geräten eingesetzt.

(2) Der Datenprotokollierungs-Statusfile kann nur durch den MicroLogix 1500 1764-LRP-Prozessor verwendet werden.

(3) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(4) Der F-File ist nur für MicroLogix 1200- und 1500-Steuerungen der Serie C und höher gültig.

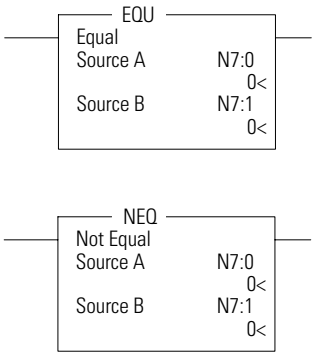
(5) Verwenden Sie den Hochgeschwindigkeits-Zählerakkumulator (HSC.ACC) nur für Quelle A in den GRT-, LES-, GEQ- und LEQ-Befehlen.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

Wenn mindestens einer der Operanden ein Fließkomma-Wert ist, gilt:

- Für EQU, GEQ, GRT, LEQ und LES – Wenn keine der Quellen eine Nummer ist (NAN), ändert sich der Status des Strompfades zu unwahr.
- Für NEQ – Wenn keine der Quellen eine Nummer ist (NAN), bleibt der Status des Strompfades wahr.

# EQU – Gleich
# NEQ – Ungleich




Befehlstyp: Eingang

**Tabelle 9.2 Ausführungszeit für die Befehle EQU und NEQ**

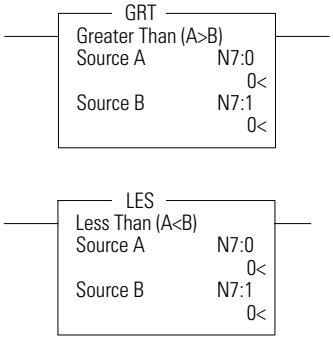
| Steuerung | Befehl | Datengröße | Strompfad | |
|---|---|---|---|---|
| | | | wahr | unwahr |
| MicroLogix 1200 | EQU | Wort | 1,3 µs | 1,1 µs |
| | | Doppelwort | 2,8 µs | 1,9 µs |
| | NEQ | Wort | 1,3 µs | 1,1 µs |
| | | Doppelwort | 2,5 µs | 2,7 µs |
| MicroLogix 1500 | EQU | Wort | 1,2 µs | 1,1 µs |
| | | Doppelwort | 2,6 µs | 1,9 µs |
| | NEQ | Wort | 1,2 µs | 1,1 µs |
| | | Doppelwort | 2,3 µs | 2,5 µs |

Mit dem EQU-Befehl können Sie prüfen, ob ein Wert gleich einem zweiten Wert ist. Mit dem NEQ-Befehl können Sie prüfen, ob ein Wert ungleich einem zweiten Wert ist.

**Tabelle 9.3 Funktionsweise der Befehle EQU und NEQ**

| Befehl | Beziehung zwischen Quellenwerten | Daraus resultierender Strompfadstatus |
|---|---|---|
| EQU | A = B | wahr |
| | A ≠ B | unwahr |
| NEQ | A = B | unwahr |
| | A ≠ B | wahr |

# GRT – Größer als
# LES – Kleiner als




Befehlstyp: Eingang

**Tabelle 9.4 Ausführungszeit für die Befehle GRT und LES**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 1,3 µs | 1,1 µs |
| | Doppelwort | 2,8 µs | 2,7 µs |
| MicroLogix 1500 | Wort | 1,2 µs | 1,1 µs |
| | Doppelwort | 2,6 µs | 2,5 µs |

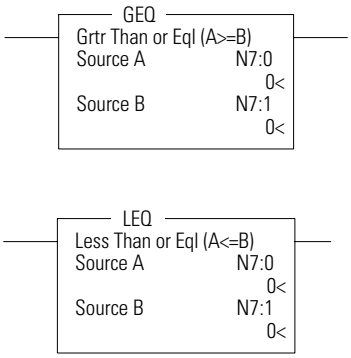
Mit dem GRT-Befehl können Sie prüfen, ob ein Wert größer als ein zweiter Wert ist. Mit dem LES-Befehl können Sie prüfen, ob ein Wert kleiner als ein zweiter Wert ist.

**Tabelle 9.5 Funktionsweise der Befehle GRT und LES**

| Befehl | Beziehung zwischen Quellenwerten | Daraus resultierender Strompfadstatus |
|---|---|---|
| GRT | A > B | wahr |
| | A ≤ B | unwahr |
| LES | A ≥ B | unwahr |
| | A < B | wahr |

**WICHTIG** Verwenden Sie den Hochgeschwindigkeits-Zählerakkumulator (HSC.ACC) nur für Quelle A in den GRT-, LES-, GEQ- und LEQ-Befehlen.

# GEQ – Größer als oder gleich
# LEQ – Kleiner als oder gleich




Befehlstyp: Eingang

**Tabelle 9.6 Ausführungszeit für die Befehle GEQ und LEQ**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 1,3 µs | 1,1 µs |
| | Doppelwort | 2,8 µs | 2,7 µs |
| MicroLogix 1500 | Wort | 1,2 µs | 1,1 µs |
| | Doppelwort | 2,6 µs | 2,5 µs |

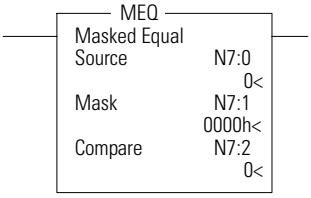
Mit dem GEQ-Befehl können Sie prüfen, ob ein Wert größer als oder gleich einem zweiten Wert ist. Mit dem LEQ-Befehl können Sie prüfen, ob ein Wert kleiner als oder gleich einem zweiten Wert ist.

**Tabelle 9.7 Funktionsweise der Befehle GEQ und LEQ**

| Befehl | Beziehung zwischen Quellenwerten | Daraus resultierender Strompfadstatus |
|---|---|---|
| GEQ | A ≥ B | wahr |
| | A < B | unwahr |
| LEQ | A > B | unwahr |
| | A ≤ B | wahr |

**WICHTIG** Verwenden Sie den Hochgeschwindigkeits-Zählerakkumulator (HSC.ACC) nur für Quelle A in den GRT-, LES-, GEQ- und LEQ-Befehlen.

# MEQ – Maskierter Vergleich auf gleich




Befehlstyp: Eingang

**Tabelle 9.8 Ausführungszeit für MEQ-Befehl**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 1,9 µs | 1,8 µs |
| | Doppelwort | 3,9 µs | 3,1 µs |
| MicroLogix 1500 | Wort | 1,7 µs | 1,7 µs |
| | Doppelwort | 3,5 µs | 2,9 µs |

Mit dem MEQ-Befehl können Sie durch ein Maske prüfen, ob ein Wert (Quellenadresse) gleich einem zweiten Wert (Bezugsadresse) ist. Die Quellenadresse und die Bezugsadresse werden mit einem logischen AND-Befehl mit der Maske verbunden. Dann werden die Ergebnisse verglichen. Wenn die resultierenden Werte gleich sind, ist der Strompfad wahr. Wenn die resultierenden Werte nicht gleich sind, ist der Strompfad unwahr.

Beispiel:

| Quelle: | | | | | | | | | | | | | | | | Bezugsadresse: | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Maske: | | | | | | | | | | | | | | | | Maske: | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| Zwischenergebnis: | | | | | | | | | | | | | | | | Zwischenergebnis: | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Vergleich der Zwischenergebnisse: ungleich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Der Quellenwert, der Bezugswert und die Maske müssen dieselben Datengröße (Wort oder Doppelwort) aufweisen. Für die Maske und den Bezugswert gelten folgende Bereiche:

- -32768 bis 32767 (Wort)
- -2.147.483.648 bis 2.147.483.647 (Doppelwort)

Die Maske wird als hexadezimaler Wert ohne Vorzeichen im Bereich von 0000 bis FFFF FFFF dargestellt.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 9.9 Gültige Adressierungsmodi und Filetpyen für MEQ-Befehl**
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles(1) | | | | | | | | | | | | Adressierungsmodus(3) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | CS - Komm | IOS - E/A | DLS - Datenprotokoll(2) | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | | • | • | |
| Maske | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Bezugswert | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |

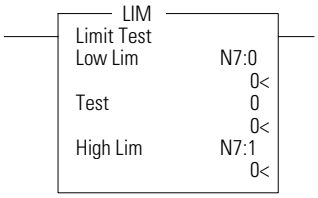
(1) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur zur Verwendung mit MicroLogix 1200- und 1500 BXB-Geräten eingesetzt.

(2) Der Datenprotokollierungs-Statusfile kann nur durch den MicroLogix 1500 1764-LRP-Prozessor verwendet werden.

(3) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

## LIM – Grenzwerttest




Befehlstyp: Eingang

**Tabelle 9.10 Ausführungszeit für LIM-Befehl**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 6,4 µs | 6,1 µs |
| | Doppelwort | 14,4 µs | 13,6 µs |
| MicroLogix 1500 | Wort | 5,5 µs | 5,3 µs |
| | Doppelwort | 12,2 µs | 11,7 µs |

Mit dem LIM-Befehl können Sie prüfen, ob Werte innerhalb eines angegebenen Bereichs liegen. Der LIM-Befehl wird anhand der Werte für Untergrenze, Prüfwert und Obergrenze durchgeführt (siehe folgende Tabelle).

**Tabelle 9.11 Funktionsweise des LIM-Befehls mit Untergrenze, Prüfwert und Obergrenze**

| Wenn: | Und: | Strompfad-status |
|---|---|---|
| Untergrenze ≤ Obergrenze | Untergrenze ≤ Test ≤ Obergrenze | wahr |
| Untergrenze ≤ Obergrenze | Prüfwert < Untergrenze oder Prüfwert > Obergrenze | unwahr |
| Obergrenze < Untergrenze | Obergrenze < Test < Untergrenze | unwahr |
| Obergrenze < Untergrenze | Test ≥ Obergrenze oder Test ≤ Untergrenze | wahr |

Die Werte für Untergrenze, Prüfwert und Obergrenze können Wortadressen oder Konstanten sein, wobei für Kombinationen die folgenden Einschränkungen gelten:

- Wenn der Prüfwert eine Konstante ist, müssen sowohl für die Unter- wie auch die Obergrenze Wort- oder Doppelwortadressen eingegeben werden.
- Wenn der Prüfwert eine Wort- oder Doppelwortadresse ist, können als Unter- und als Obergrenze eine Konstante, eine Wort- oder eine Doppelwortadresse angegeben werden. Allerdings darf nicht für beide Grenzen eine Konstante angegeben werden.

Wenn Sie Werte unterschiedlicher Größe verwenden, werden alle diese Werte in das Format des größten Wertes umgewandelt. Wenn beispielsweise ein Wort und ein Doppelwort verwendet werden, wird das Wort in ein Doppelwort umgewandelt.

Gültige Datenbereiche:

- -32768 bis 32767 (Wort)
- -2.147.483.648 bis 2.147.483.647 (Doppelwort)

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 9.12 Gültige Adressierungsmodi und Filetypen für LIM-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles(1) | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll(2) | Adressierungsmodus(3) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F(4) | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Untergrenze | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Prüfwert | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Obergrenze | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |

(1) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur zur Verwendung mit MicroLogix 1200- und 1500 BXB-Geräten eingesetzt.

(2) Der Datenprotokollierungs-Statusfile kann nur durch den MicroLogix 1500 1764-LRP-Prozessor verwendet werden.

(3) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(4) Der F-File ist nur für MicroLogix 1200- und 1500-Steuerungen der Serie C und höher gültig.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# Mathematische Befehle

## Allgemeine Informationen

Machen Sie sich zuerst mit den folgenden Themen am Anfang dieses Kapitels vertraut, bevor Sie die mathematischen Befehle verwenden:

- „Mathematische Befehle verwenden“
- „Aktualisierung der mathematischen Status-Bits“
- „Fließkomma-Datenfile (F) verwenden“

## Befehle

Mit diesen Ausgangsbefehlen können Sie einen Ausdruck oder einen bestimmten mathematischen Befehl für Berechnungen verwenden.

| Befehl | Zweck | Seite |
|---|---|---|
| ADD - Addition | Addition zweier Werte | 10-7 |
| SUB - Subtraktion | Subtraktion zweier Werte | 10-7 |
| MUL - Multiplikation | Multiplikation zweier Werte | 10-8 |
| DIV - Division | Division eines Wertes durch einen anderen | 10-8 |
| NEG - Negation | Änderung des Vorzeichens des Quellenwerts und Einfügung in das Ziel | 10-9 |
| CLR - Löschen | Rücksetzen aller Bits in einem Wort auf null | 10-9 |
| ABS - Absolutwert | Berechnen des Absolutwertes eines Quellenwertes | 10-10 |
| SQR - Quadratwurzel | Berechnung der Quadratwurzel eines Werts | 10-15 |
| SCL - Skalierung | Skalieren eines Werts | 10-12 |
| SCP - Skalierung mit Parametern | Skalieren eines Werts auf einen Bereich, der durch eine lineare Beziehung festgelegt wird | 10-13 |

## Mathematische Befehle verwenden

Die meisten mathematischen Befehle werden mit drei Parametern verwendet, Quelle A, Quelle B und Ziel (zusätzliche erforderliche Parameter werden an den entsprechenden Stellen in diesem Kapitel beschrieben). Die mathematische Operation wird mit beiden Quellenwerten durchgeführt. Das Ergebnis wird im Ziel gespeichert.

Bei der Verwendung mathematischer Befehle sind folgende Aspekte zu beachten:

- Die Quelle und das Ziel können unterschiedliche Datengrößen auf- weisen. Quellen werden mit der höchsten Genauigkeit (Wort oder Doppelwort) der Operanden bewertet. Dann wird das Ergebnis umgewandelt und an die Größe des Ziels angepasst. Wenn im Ziel kein ausreichender Speicherplatz für den Quellenwert mit Vor- zeichen vorhanden ist, wird der Überlauf wie folgt gehandhabt:
  - Wenn das Bit zur Auswahl der mathematischen Überlauffunk- tion nicht gesetzt ist, wird ein gesättigtes Ergebnis im Ziel gespeichert. Wenn die Quelle positiv ist, lautet das Ziel +32767 (Wort) oder +2.147.483.647 (Doppelwort). Wenn das Ergebnis negativ ist, lautet das Ziel -32768 (Wort) oder -2.147.483.648 (Doppelwort).
  - Wenn das Bit zur Auswahl der mathematischen Überlauf- funktion gesetzt ist, wird der gekürzte Quellenwert ohne Vorzeichen im Ziel gespeichert.
- Bei den Quellen kann es sich um Konstanten oder Adressen handeln; allerdings darf nicht für beide Quellen eine Konstante angegeben werden.
- Gültige Bereiche für Konstanten: -32768 bis 32767 (Wort) und -2.147.483.648 bis 2.147.483.647 (Doppelwort).

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 10.1 Gültige Adressierungsmodi und Filetypen für die mathematischen Befehle (ADD, SUB, MUL, DIV, NEG, CLR)**
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles[1] | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll[2] | Adressierungsmodus[3] | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F[4] | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle A | • | • | • | • | • | • | • | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Quelle B | • | • | • | • | • | • | • | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Ziel | • | • | • | • | • | • | • | | • | • | | • | • | • | • | • | | | • | | | | | | • | • | | • | • | |

(1) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur zur Verwendung mit MicroLogix 1200- und 1500 BXB-Geräten eingesetzt.

(2) Der Datenprotokollierungs-Statusfile kann nur durch den MicroLogix 1500 1764-LRP-Prozessor für die folgenden mathematischen Befehle verwendet werden. ADD, SUB, MUL, DIV, NEG und SCP.

(3) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(4) Der F-File ist nur für MicroLogix 1200- und 1500-Steuerungen der Serie C und höher gültig.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

## Aktualisierung der mathematischen Status-Bits

Nach Ausführung eines mathematischen Befehls werden die mathematischen Status-Bits im Statusfile aktualisiert. Die mathematischen Status-Bits befinden sich in Wort 0 des Prozessor-Statusfiles.

**Tabelle 10.2 Mathematische Status-Bits**

| Bit: | | Aktion des Prozessors: |
|---|---|---|
| **S:0/0** | Übertrag | das Bit wird gesetzt, wenn ein Übertrag generiert wird; andernfalls wird das Bit rückgesetzt |
| **S:0/1** | Überlauf | das Bit wird gesetzt, wenn im Ziel kein ausreichender Speicherplatz für das Ergebnis eines mathematischen Befehls vorhanden ist; andernfalls wird das Bit rückgesetzt |
| **S:0/2** | Null-Bit | das Bit wird gesetzt, wenn das Ergebnis null ist; andernfalls wird das Bit rückgesetzt |
| **S:0/3** | Vorzeichen-Bit | das Bit wird gesetzt, wenn das Ergebnis negativ ist (MSB ist gesetzt); andernfalls wird das Bit rückgesetzt |
| **S:2/14** | Ausgewählte mathematische Überlauffunktion(1) | der Status dieses Bits wird geprüft, um den Wert des Ergebnisses bei einem Überlauf zu ermitteln |
| **S:5/0** | Überlauferkennung(1) | das Bit wird gesetzt, wenn das Überlauf-Bit gesetzt wird; andernfalls wird das Bit rückgesetzt |

(1) Steuer-Bits.

## Überlauferkennungs-Bit, S:5/0

Dieses Bit für geringfügige Störungen (S:5/0) wird nach Erkennen eines mathematischen Überlaufs oder einer Division durch 0 gesetzt. Wird dieses Bit nach der Ausführung eines END- oder TND-Befehls gesetzt, zeigt der Prozessor einen behebbaren schwerwiegenden Fehler an (Fehlercode 0020).

Bei Anwendungen, in denen ein mathematischer Überlauf oder eine Division durch Null vorkommen, können Steuerungsfehler durch Verwendung eines OTU-Befehls (Ausgang rücksetzen) mit der Adresse S:5/0 verhindert werden. Der Strompfad muss zwischen dem Überlaufpunkt und dem END- oder TND-Befehl liegen.

Die folgende Abbildung zeigt den Strompfad, der zum Rücksetzen des Überlaufauffang-Bits verwendet werden kann.

# Fließkomma-Datenfile (F) verwenden

## Filebeschreibung

Fließkomma-Files enthalten IEEE-754-Fließkomma-Datenelemente. Im Folgenden wird ein Fließkomma-Element dargestellt. In jedem Fließkomma-File können bis zu 256 dieser Elemente enthalten sein.

**Tabelle 10.3 Struktur eines Fließkomma-Datenfiles**

| **Fließkomma-Element** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** | **15** | **14** | **13** | **12** | **11** | **10** | **09** | **08** | **07** | **06** | **05** | **04** | **03** | **02** | **01** | **00** |
| S(1) | Wert des Exponenten | | | | | | | | Mantisse | | | | | | | | | | | | | | | | | | | | | | |
| Höherwertiges Wort | | | | | | | | | | | | | | | | Niederwertiges Wort | | | | | | | | | | | | | | |

(1) S = Vorzeichen-Bit

Fließkomma-Nummern werden im IEEE-754-Format angegeben; dabei gilt:

- Bit 31 ist das Vorzeichen-Bit. Dieses Bit wird für negative Nummern gesetzt (beachten Sie, dass Negativ-Null ein gültiger Wert ist).
- Bits 23 bis 30 sind die Exponenten.
- Bits 0 bis 22 sind die Mantissen.

Der Wert, der von einer 32-Bit-Fließkomma-Nummer dargestellt wird (keiner der Ausnahmewerte, die auf Seite 10-5 festgelegt wurden), wird durch den folgenden Ausdruck wiedergegeben. Beachten Sie die Wiederherstellung des unterdrückten und bedeutendsten Bits der Mantisse.

$$(-1)^{s} \times 2^{e-127} \times (1 + m)$$

Wobei:

*s* für das Vorzeichen-Bit (0 oder 1) steht

*e* für den Exponenten (1 bis 254) steht

*m* für die Mantisse ($0 \leq f < 1$) steht

Der gültige Bereich für die Fließkomma-Nummer reicht von $-3{,}4028 \times 10^{38}$ bis $+3{,}4028 \times 10^{38}$.

## Definitionen

**Überlauf** – Ein Überlauf tritt auf, wenn das Ergebnis einer Operation einen Exponenten hervorruft, dessen Wert größer als 254 ist.

**Unterlauf** – Ein Unterlauf tritt auf, wenn das Ergebnis einer Operation einen Exponenten hervorruft, dessen Wert kleiner als 1 ist.

## Ausnahmewerte des Fließkommas

**Null** – Die Null wird durch einen Exponenten und eine Mantisse mit den Werten null dargestellt. Es sind sowohl Positiv- als auch Negativ-Null gültig.

**Denormalisiert** – Dies wird durch einen Exponenten mit dem Wert null und einer Mantisse mit einem Wert ungleich null dargestellt. Da denormalisierte Nummern einen sehr kleinen und unbedeutenden Wert annehmen, werden sie als null betrachtet, wenn sie für die meisten Operationen als Quellenoperand eingesetzt werden. Dies verkürzt die Ausführungszeit. Denormalisierte Nummern werden nicht mihilfe von Befehlen erstellt, sie werden jedoch durch einige Befehle übermittelt. Null wird durch einen Unterlauf erzeugt.

**Unendlich** – Dies wird durch einen Exponenten mit dem Wert 255 und einer Mantisse mit dem Wert null dargestellt. Es werden sowohl positive als auch negative unendliche Werte erzeugt, wenn Operationen überlaufen. Die Unendlichkeit wird durch Berechnungen übermittelt.

**NAN (Not a number; keine Nummer)** – Dies wird durch einen Exponenten mit dem Wert 255 und einer Mantisse mit einem Wert ungleich null dargestellt. NANs werden zur Kennzeichnung von Ergebnissen verwendet, die mathematisch undefinierbar sind, wie z. B. 0/0 oder die Addition von positiv unendlich mit negativ unendlich. Alle Operationen, denen NAN als Eingang zugeordnet werden, müssen eine NAN als Ausgang erzeugen.

## LSB-Regel zum Runden auf gerade Zahlen

Fließkomma-Operationen werden mithilfe der Regel zum Runden auf gerade Zahlen gerundet. Wenn die Bits des Ergebnisses zur Rechten des unbedeutendsten Bits (LSB) einen Wert darstellen, der kleiner ist als die Hälfte des LSB, bleibt das Ergebnis bestehen. Wenn die Bits zur Rechten des LSBs einen Wert darstellen, der größer ist als die Hälfte des LSB, wird das Ergebnis aufgerundet, indem ein LSB dazu addiert wird. Wenn die Bits zur Rechten des LSB einen Wert darstellen, der genau der Hälfte des LSB entspricht, wird der Wert auf- oder abgerundet, sodass der LSB eine gerade Zahl darstellt.

## Adressierung von Fließkomma-Files

Nachfolgend wird das Adressierungsformat für Fließkomma-Datenfiles dargestellt.

| Format | Bedeutung | | |
|---|---|---|---|
| **Ff:e** | **F** | Fließkomma-File | |
| | **f** | Filenummer | Die Filenummer kann aus einem Bereich von 8 (Standard) und 255 ausgewählt werden. |
| | **:** | Elementendezeichen | |
| | **e** | Elementnummer | Die Elementnummer kann aus einem Bereich von 0 bis 255 ausgewählt werden. |
| **Beispiele:** | **F8:2**<br>**F10:36** | | **Fließkommafile 8, Element 2**<br>**Fließkommafile 10, Element 36** |

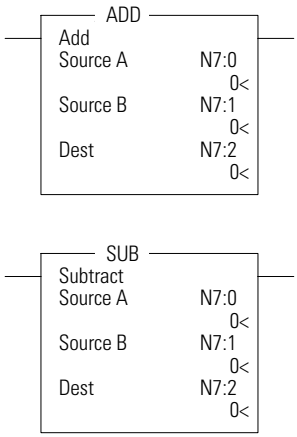
## Fließkomma-Werte programmieren

In der folgenden Tabelle werden Punkte aufgeführt, die Sie bei der Arbeit mit Fließkomma-Daten beachten sollten.

**WICHTIG** Diese Regeln gelten nicht für den SCP-Befehl. Regeln für diesen Befehl finden Sie auf Seite 10-14.

| **Überlegungen für den Umgang mit Fließkomma-Daten** |
|---|
| Wenn mindestens einer der Operanden ein Fließkomma-Wert ist:<br>• Wenn eine der Quellen NAN ist, ist das Ergebnis auch NAN.<br>• Alle Überläufe haben undendliche Ergebnisse mit dem entsprechen Vorzeichen.<br>• Alle Unterläufe haben als Ergebnis Positiv-Null.<br>• Alle denormalisierten Quelldaten werden als Positiv-Null behandelt.<br>• Ergebnisse werden immer mithilfe der Regel zum Runden auf gerade Zahlen gerundet.<br>• Wenn das Ziel eine Ganzzahl und das Ergebnis NAN oder unendlich ist, wird ein gesättigtes Ergebnis (–32768 oder +32767 für Wort oder –2 147 836 648 oder +2 147 836 647 für Doppelwort) im Ziel gespeichert und das Bit zur Auswahl der mathematischen Überlauffunktion ignoriert.<br>• Wenn das Ziel eine Ganzzahl ist, wird das gerundete Ergebnis gespeichert. Wenn nach dem Runden ein Überlauf eintritt, wird ein gesättigtes Ergebnis im Ziel gespeichert und das Bit zur Auswahl der mathematischen Überlauffunktion ignoriert. Die gesättigten Ergebnisse sind:<br>– Wenn das Ziel eine Ganzzahl und das Ergebnis positiv ist, beträgt das Überlauf-Ziel +32767 (Wort) oder +2 147 483 648 (Doppelwort).<br>– Wenn das Ziel eine Ganzzahl und das Ergebnis negativ ist, beträgt das Überlauf-Ziel –32767 (Wort) oder –2 147 483 648 (Doppelwort). |
| Mathematische Status-Bit-Updates:<br>• Übertrag – Das Bit wird rückgesetzt.<br>• Überlauf – Das Bit ist gesetzt, wenn das Ergebnis unendlich oder NAN ist oder wenn eine Konvertierung zu einer Ganzzahl überläuft; andernfalls wird es rückgesetzt.<br>• Null – Das Bit ist gesetzt, wenn niederwertige Bits 31der Fließkomma-Daten alle null als Ergebnis haben; andernfalls wird es rückgesetzt.<br>• Vorzeichen – das Bit ist gesetzt, wenn das bedeutendste Bit des Ziels gesetzt ist (Bit 15 für Wort, Bit 31 für Doppelwort oder Fließkomma-Daten); andernfalls wird es rückgesetzt.<br>• Überlauferkennung – das mathematische Überlauferkennungs-Bit wird nur gesetzt, wenn auch das Überlauf-Bit gesetzt ist; andernfalls verbleibt das Bit im vorherigen Status. |

# ADD – Addition
# SUB – Subtraktion




Befehlstyp: Ausgang

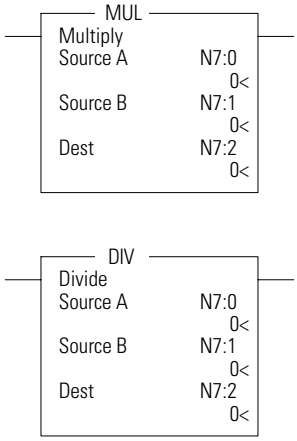
**Tabelle 10.4 Ausführungszeit für die Befehle ADD und SUB**

| Steuerung | Befehl | Datengröße | Strompfad | |
|---|---|---|---|---|
| | | | wahr | unwahr |
| MicroLogix 1200 | ADD | Wort | 2,7 µs | 0,0 µs |
| | | Doppelwort | 11,9 µs | 0,0 µs |
| | SUB | Wort | 3,4 µs | 0,0 µs |
| | | Doppelwort | 12,9 µs | 0,0 µs |
| MicroLogix 1500 | ADD | Wort | 2,5 µs | 0,0 µs |
| | | Doppelwort | 10,4 µs | 0,0 µs |
| | SUB | Wort | 2,9 µs | 0,0 µs |
| | | Doppelwort | 11,2 µs | 0,0 µs |

Verwenden Sie den ADD-Befehl, um einen Wert zu einem anderen zu addieren (Quelle A + Quelle B) und die Summe in die Zieladresse zu stellen.

Verwenden Sie den SUB-Befehl, um einen Wert zu einem anderen abzuziehen (Quelle A - Quelle B) und die Summe in die Zieladresse zu stellen.

# MUL – Multiplikation
# DIV – Division




Befehlstyp: Ausgang

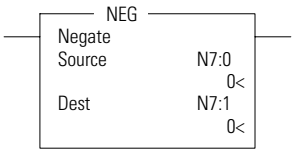
**Tabelle 10.5 Ausführungszeit für die Befehle MUL und DIV**

| Steuerung | Befehl | Datengröße | Strompfad | |
|---|---|---|---|---|
| | | | wahr | unwahr |
| MicroLogix 1200 | MUL | Wort | 6,8 µs | 0,0 µs |
| | | Doppelwort | 31,9 µs | 0,0 µs |
| | DIV | Wort | 12,2 µs | 0,0 µs |
| | | Doppelwort | 42,8 µs | 0,0 µs |
| MicroLogix 1500 | MUL | Wort | 5,8 µs | 0,0 µs |
| | | Doppelwort | 27,6 µs | 0,1 µs |
| | DIV | Wort | 10,3 µs | 0,0 µs |
| | | Doppelwort | 36,7 µs | 0,0 µs |

Verwenden Sie den MUL-Befehl, um einen Wert mit einem anderen zu multiplizieren (Quelle A x Quelle B) und das Ergebnis in die Zieladresse zu stellen.

Verwenden Sie den DIV-Befehl, um einen Wert durch einen anderen zu teilen (Quelle A/Quelle B) und das Ergebnis in die Zieladresse zu stellen. Wenn die Quellen einzelne Wörter darstellen und die Zieladresse direkt auf S:13 (mathematisches Register) adressiert ist, werden der Quotient in S:14 und der verbleibende Wert in S:13 gespeichert. Bei der Verwendung von Doppelworten werden die Ergebnisse gerundet.

# NEG – Negation



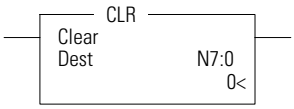

Befehlstyp: Ausgang

**Tabelle 10.6 Ausführungszeit des NEG-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 2,9 µs | 0,0 µs |
| | Doppelwort | 12,1 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 1,9 µs | 0,0 µs |
| | Doppelwort | 10,4 µs | 0,0 µs |

Verwenden Sie den NEG-Befehl, um das Vorzeichen der Quelle zu ändern und das Ergebnis in die Zieladresse zu stellen.

# CLR – Löschen




Befehlstyp: Ausgang

**Tabelle 10.7 Ausführungszeit des ENC-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 1,3 µs | 0,0 µs |
| | Doppelwort | 6,3 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 1,2 µs | 0,0 µs |
| | Doppelwort | 5,5 µs | 0,0 µs |

Verwenden Sie den CLR-Befehl, um das Ziel auf den Wert Null zu setzen.

# ABS – Absolutwert

Befehlstyp: Ausgang

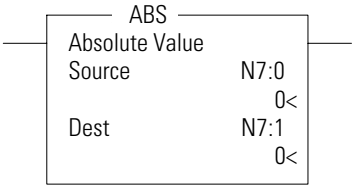



**Tabelle 10.8 Ausführungszeit des ABS-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 3,8 µs | 0,0 µs |
| MicroLogix 1500 | 3,1 µs | 0,0 µs |

Mit dem ABS-Befehl wird der Absolutwert der Quelle in das Ziel verschoben. Der Datenbereich für diesen Befehl beträgt
–2 147 483 648 bis 2 147 483 647 oder IEEE-754-Fließkomma-Wert.

Quelle und Ziel müssen nicht vom gleiche Datentyp sein. Wenn das Ergebnis mit Vorzeichen jedoch nicht in das Ziel passt, geschieht Folgendes.

**Tabelle 10.9 ABS-Ergebnis passt nicht in Ziel**

| Beide Operanden sind Ganzzahlen | Mindestens ein Operand gehört den Fließkomma-Daten an |
|---|---|
| • Wenn das Bit zur Auswahl der mathematischen Überlauffunktion gelöscht wurde, wird ein gesättigtes Ergebnis (32767 für Wort oder 2 147 836 647 für Doppelwort) im Ziel gespeichert.<br>• Wenn das Bit zur Auswahl der mathematischen Überlauffunktion gesetzt wurde, wird der abgeschnittene Wert des Ergebnisses ohne Vorzeichen im Ziel gespeichert. | • Mit dem ABS-Befehl wird das Vorzeichen-Bit gelöscht. Mit den verbleibenden Bits wird keine Operation durchgeführt.<br>• Wenn das Ziel eine Ganzzahl und das Ergebnis NAN oder unendlich ist, wird ein gesättigtes Ergebnis (32767 für Wort oder 2 147 836 647 für Doppelwort) im Ziel gespeichert und das Bit zur Auswahl der mathematischen Überlauffunktion ignoriert.<br>• Wenn das Ziel eine Ganzzahl ist, wird das gerundete Ergebnis gespeichert. Wenn nach dem Runden ein Überlauf eintritt, wird ein gesättigtes Ergebnis (32767 für Wort oder 2 147 836 647 für Doppelwort) im Ziel gespeichert und das Bit zur Auswahl der mathematischen Überlauffunktion ignoriert. |

In der folgenden Tabelle wird dargestellt, wie mathematische Statusbits durch die Ausführung des ABS-Befehls aktualisiert werden:

**Tabelle 10.10 Aktualisierung der mathematischen Status-Bits**

| Beide Operanden sind Ganzzahlen | Mindestens ein Operand gehört den Fließkommadaten an |
|---|---|
| • Übertrag - Das Bit wird gesetzt, wenn der Eingang negativ ist; anderfalls wird es rückgesetzt.<br>• Überlauf - Das Bit wird gesetzt, wenn das Ergebnis mit Vorzeichen nicht ins Ziel passt; andernfalls wird es rückgesetzt.<br>• Null - Das Bit wird gesetzt, wenn das Ziel nur aus null besteht; andernfalls wird es rückgesetzt.<br>• Vorzeichen - Das Bit wird gesetzt, wenn das bedeutendste Bit des Ziels gesetzt ist; andernfalls wird es rückgesetzt.<br>• Überlauferkennung - Das mathematische Überlauferkennungs-Bit wird nur gesetzt, wenn das Überlauf-Bit gesetzt ist. Andernfalls verbleibt das Bit in seinem letzten Zustand. | • Übertrag - Das Bit wird rückgesetzt.<br>• Überlauf - Das Bit wird gesetzt, wenn das Ergebnis mit Vorzeichen unendlich oder NAN ist oder wenn es nicht ins Ziel passt; andernfalls wird es rückgesetzt.<br>• Null - Das Bit wird gesetzt, wenn das Ziel nur aus null besteht; andernfalls wird es rückgesetzt.<br>• Vorzeichen - Das Bit wird gesetzt, wenn das bedeutendste Bit des Ziels gesetzt ist; andernfalls wird es rückgesetzt.<br>• Überlauferkennung - Das mathematische Überlauferkennungs-Bit wird nur gesetzt, wenn auch das Überlauf-Bit gesetzt ist. Andernfalls verbleibt das Bit im vorherigen Status. |

Adressierungsmodi und Filetypen werden in der folgenden Tabelle dargestellt:

**Tabelle 10.11 Gültige Adressierungsmodi und Filetypen für ABS-Befehl**

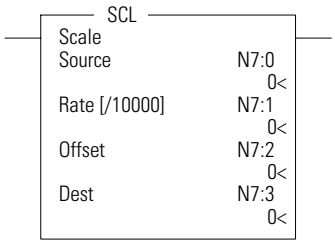
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | | | | Adressierungsmodus[1] | | | Adressierungsebene | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | CS - Komm | IOS - E/A | DLS - Datenprotokoll | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Fließkomma | Element |
| Quelle | • | • | • | • | • | • | • | | • | • | • | • | • | • | • | • | • | • | | • | • | • | | • | • | • | | • | • | • | |
| Ziel | • | • | • | • | • | • | • | | • | • | • | • | • | • | • | • | | | | | | | | | • | • | | • | • | • | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# SCL – Skalierung




Befehlstyp: Ausgang

**Tabelle 10.12 Ausführungszeit des SCL-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 10,5 µs | 0,0 µs |
| MicroLogix 1500 | 8,7 µs | 0,0 µs |

Mit dem SCL-Befehl wird der Wert der Quellenadresse mit der angegebenen Steigung (Rate) multipliziert. Das Ergebnis wird dann zum Offset addiert, und das gerundete Ergebnis wird in das Ziel gestellt.

Die folgenden Gleichungen drücken die lineare Beziehung zwischen dem Eingangswert und dem daraus resultierenden skalierten Wert aus:

skalierter Wert = [(Rate x Quelle)/10000] + Offset, wobei

- Rate = (skalierter max. Wert – skalierter min. Wert)/(Eingang max. Wert – Eingang min. Wert)
- Offset = skalierter min. Wert – (Eingang min. Wert x Rate)

Bei der Rate und dem Offset kann es sich jeweils um unmittelbare Werte handeln. Gültiger Bereich für Rate und Offset: -32768 bis 32767.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 10.13 Gültige Adressierungsmodi und Filetypen für SCL-Befehl**
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

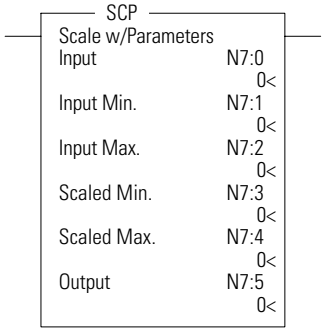
| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komms | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | | | | | | | | | | | | | | | | | | • | • | | • | | |
| Rate | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | • | • | | • | | |
| Versatz | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | • | • | | • | | |
| Ziel | • | • | | • | • | • | | | | | | | | | | | | | | | | | | | • | • | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

**WICHTIG** Verwenden Sie den Hochgeschwindigkeits-Zählerakkumulator (HSC.ACC) nicht für den Zieladressenparameter im SCL-Befehl.

# SCP – Skalierung mit Parametern




Befehlstyp: Ausgang

**Tabelle 10.14 Ausführungszeit des SCP-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 31,5 µs | 0,0 µs |
| | Doppelwort | 52,2 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 27,0 µs | 0,0 µs |
| | Doppelwort | 44,7 µs | 0,0 µs |

Mit dem SCP-Befehl wird ein skalierter Ausgangswert erzeugt, der in linearer Beziehung zwischen dem Eingang und dem skalierten Wert steht. Dieser Befehl löst die nachfolgende Gleichung zur Ermittlung des skalierten Ausgangs auf.

$$y = [(y_1 - y_0)/(x_1 - x_0)](x - x_0) + y_0$$

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 10.15 Gültige Adressierungsmodi und Filetypen für SCP-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles(1) | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(2) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Eingang (x) | • | • | • | • | • | • | • | | • | • | | • | • | • | • | • | • | • | • | • | • | • | | | • | • | | • | • | |
| Eingang Min. ($x_0$) | • | • | | • | • | • | • | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Eingang Max. ($x_1$) | • | • | | • | • | • | • | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Skalierter min. Wert ($y_0$) | • | • | | • | • | • | • | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Skalierter max. Wert ($y_1$) | • | • | | • | • | • | • | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Ausgang (y) | • | • | • | • | • | • | • | | • | • | | • | | • | • | • | | | • | | | | | | • | • | | • | • | |

(1) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur zur Verwendung mit MicroLogix 1200- und 1500 BXB-Geräten verwendet.

(2) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

**WICHTIG** Verwenden Sie den Hochgeschwindigkeits Zählerakkumulator (HSC.ACC) nicht für den Parameter des skalierten Ausgangs im SCP-Befehl.

## Besondere Überlegungen für die Arbeit mit Fließkomma-Parametern

Wenn einer der Parameter (außer Ausgang) NAN (not a number; keine Nummer), unendlich oder denormalisiert ist, ist das Ergebnis –NAN.

Wenn $y_1 – y_0$ oder $x_1 – x_0$ in einem Überlauf resultieren, ist das Ergebnis – NAN.

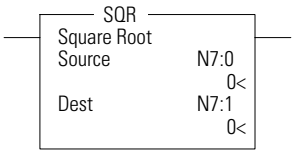
## Weitere Überlegungen

Wenn $y_1 – y_0 = 0$, wird das Ergebnis zum skalierten Anfangswert.

Wenn $x_1 – x_0 = 0$ und $x = x_0$, wird das Ergebnis zum skalierten Anfangswert.

Wenn $x_1 – x_0 = 0$ und x ist ungleich $x_0$, wird das Ergebnis zu einem negativen Überlauf (für Ganzzahlen) oder zu negativer NAN (für Fließkommawerte).

# SQR – Quadratwurzel




Befehlstyp: Ausgang

**Tabelle 10.16 Ausführungszeit des SQR-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 26,0 µs | 0,0 µs |
| | Doppelwort | 30,9 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 22,3 µs | 0,0 µs |
| | Doppelwort | 26,0 µs | 0,0 µs |

Mit dem SQR-Befehl wird die Quadratwurzel des Absolutwertes der Quelle berechnet; das gerundete Ergebnis wird in der Zieladresse gespeichert.

Die Quellenwerte müssen in dem Bereich -32768 bis 32767 (Wort) oder -2.147.483.648 bis 2.147.483.647 (Doppelwort) liegen. Wenn die Quelle negativ ist, wird das Übertrag-Status-Bit gesetzt. Weitere Informationen hierzu finden Sie unter „Aktualisierung der mathematischen Status-Bits“ auf Seite 10-3.

**Tabelle 10.17 Gültige Adressierungsmodi und Filetpyen für SQR-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden“ auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | ST | F | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | • | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Ziel | • | • | | • | • | • | | • | • | | | | | | | | | | | | | | | | • | • | | • | • | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig“) zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# Konvertierungsbefehle

Die Konvertierungsbefehle führen Multiplex- und Demultiplex- operationen mit Daten sowie Konvertierungen zwischen Binär- und Dezimalwerten aus.

| Befehl | Zweck | Seite |
|---|---|---|
| DCD - 4 in 1 auf 16 dekodieren | Dekodieren eines 4-Bit-Wertes (0 bis 15) und Aktivierung des entsprechenden Bits im 16-Bit-Ziel. | 11-2 |
| ENC - Kodierung 1 auf 16 in 4 | Kodierung einer 16-Bit-Quelle in einen 4-Bit-Wert. Geht das Ziel vom niederwertigsten bis zum höchstwertigen Bit durch und sucht nach dem ersten gesetzten Bit. Die entsprechende Bitposition wird als Ganzzahlwert in das Ziel geschrieben. | 11-3 |
| FRD - BCD in Ganzzahl | Umwandeln des BCD-Quellenwertes in einen Ganzzahlwert und Speichern dieses Wertes im Ziel. | 11-4 |
| TOD - In BCD | Umwandeln des Ganzzahl-Quellenwertes in BCD-Format und Speichern dieses Wertes im Ziel. | 11-8 |

## Kodier- und Dekodierbefehle verwenden

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 11.1 Gültige Adressierungsmodi und Filetypen für Konvertierungsbefehle**

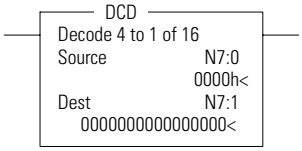
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | | | | | | | | | | | | | | | | | | • | • | | • | | |
| Ziel | • | • | | • | • | • | | | | | | | | | | | | | | | | | | | • | • | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# DCD – 4 in 1 auf 16 dekodieren




Befehlstyp: Ausgang

**Tabelle 11.2 Ausführungszeit des DCD-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 1,9 µs | 0,0 µs |
| MicroLogix 1500 | 0,9 µs | 0,0 µs |

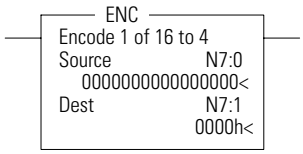
Der DCD-Befehl verwendet die ersten vier Bits des Quellenwortes, um ein Bit des Zielwortes zu setzen. Alle anderen Bits des Zielwortes werden gelöscht. Der DCD-Befehl konvertiert die Werte wie in der nachfolgenden Tabelle dargestellt:

**Tabelle 11.3 4 in 1 aus 16 dekodieren**

| Quellen-Bits | | | | | Ziel-Bits | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 15 bis 04 | 03 | 02 | 01 | 00 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| x | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| x | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| x | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| x | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| x | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| x | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| x | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

x = nicht benutzt

# ENC – Kodierung 1 auf 16 in 4




Befehlstyp: Ausgang

**Tabelle 11.4 Ausführungszeit des ENC-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 7,2 µs | 0,0 µs |
| MicroLogix 1500 | 6,8 µs | 0,0 µs |

Der ENC-Befehl geht das Ziel vom niederwertigsten bis zum höchstwertigen Bit durch und sucht nach dem ersten gesetzten Bit. Die entsprechende Bitposition wird als Ganzzahlwert in das Ziel geschrieben. Der ENC-Befehl konvertiert die Werte wie in der nachfolgenden Tabelle dargestellt:

**Tabelle 11.5 Kodieren 1 aus 16 in 4**

| Quellen-Bits | | | | | | | | | | | | | | | | Ziel-Bits | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 | 15 bis 04 | 03 | 02 | 01 | 00 |
| x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 |
| x | x | x | x | x | x | x | x | x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| x | x | x | x | x | x | x | x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| x | x | x | x | x | x | x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| x | x | x | x | x | x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| x | x | x | x | x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| x | x | x | x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| x | x | x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| x | x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| x | x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| x | x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| x | x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| x | x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| x | x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| x | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

x = bestimmt den Status des Flags

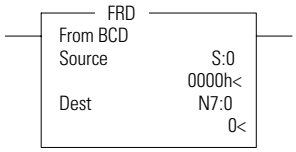
**HINWEIS** Wenn die Quelle, das Ziel und der mathematische Status Null sind, wird das Flag auf 1 gesetzt.

## Aktualisierung der mathematischen Status-Bits

**Tabelle 11.6 Mathematische Status-Bits**

| Bit: | | Aktion des Prozessors: |
|---|---|---|
| **S:0/0** | Übertrag | Das Bit wird immer rückgesetzt. |
| **S:0/1** | Überlauf | Das Bit wird gesetzt, wenn mehr als ein Bit in der Quelle gesetzt ist; andernfalls wird das Bit rückgesetzt. Das Bit für den mathematischen Überlauf (S:5/0) wird nicht gesetzt. |
| **S:0/2** | Null-Bit | Das Bit wird gesetzt, wenn das Ergebnis null ist; andernfalls wird das Bit rückgesetzt. |
| **S:0/3** | Vorzeichen-Bit | Das Bit wird immer rückgesetzt. |

# FRD – BCD in Ganzzahl




Befehlstyp: Ausgang

**Tabelle 11.7 Ausführungszeit für FRD-Befehl**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 14,1 µs | 0,0 µs |
| MicroLogix 1500 | 12,3 µs | 0,0 µs |

Mit dem FRD-Befehl wird der BCD-Quellenwert (Binary Coded Decimal - Binärdezimalcode) in einen Ganzzahlwert konvertiert; das Ergebnis wird in die Zieladresse gestellt.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 11.8 Gültige Adressierungsmodi und Filetypen für FRD-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus[1] | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | • | • | • | • | | | | | | | | | | | | | | | | | | | • | • | | • | | (2) |
| Ziel | • | • | | • | • | • | | | | | | | | | | | | | | | | | | | • | • | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Siehe „Quellenoperand FRD-Befehl" auf Seite 11-5.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

## Quellenoperand FRD-Befehl

Bei der Quelle kann es sich um eine Wortadresse oder das Rechenregister handeln. Maximale Werte für BCD-Quelle:

- 9999, wenn die Quelle eine Wortadresse ist (nur vierstelliger BCD-Wert möglich).
- 32768, wenn die Quelle das Rechenregister ist (fünfstelliger BCD-Wert möglich, erste vier Stellen in S:13 gespeichert, höhere Stellen in S:14 gespeichert).

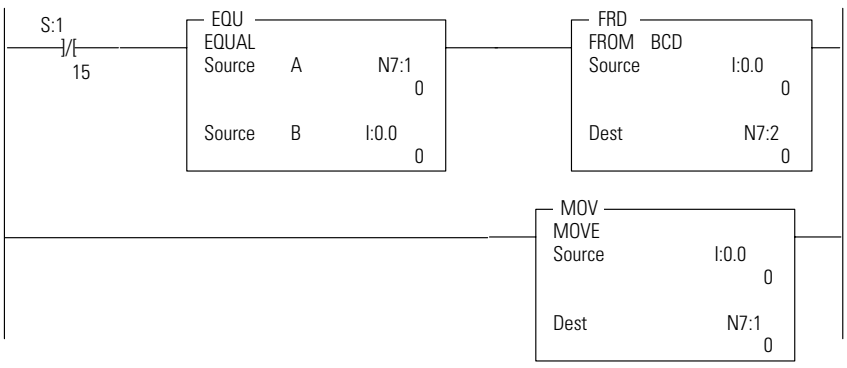
Wenn es sich bei der Quelle um das Rechenregister handelt, muss dieses direkt als S:13 adressiert werden. S:13 ist das einzige Element des Statusfiles, das verwendet werden kann.

## Aktualisierung der mathematischen Status-Bits

**Tabelle 11.9 Mathematische Status-Bits**

| Bit: | | Aktion des Prozessors: |
|---|---|---|
| **S:0/0** | Übertrag | Das Bit wird immer rückgesetzt. |
| **S:0/1** | Überlauf | Das Bit wird gesetzt, wenn der in der Quelle gespeicherte Wert kein BCD-Wert ist oder der zu konvertierende Wert größer als 32767 ist; andernfalls wird das Bit rückgesetzt. Bei einem Überlauf wird außerdem das Flag für geringfügige Störungen gesetzt. |
| **S:0/2** | Null-Bit | Das Bit wird gesetzt, wenn das Ergebnis null ist; andernfalls wird das Bit rückgesetzt. |
| **S:0/3** | Vorzeichen-Bit | Das Bit wird immer rückgesetzt. |

**HINWEIS** Es empfiehlt sich, für alle BCD-Eingangsgeräte vor der Ausführung des FRD-Befehls eine Kontakt- planfilterung durchzuführen. Bereits die geringste Abweichung in der Punkt-zu-Punkt- Eingangsfilter- verzögerung kann einen Überlauf des FRD-Befehls durch Umwandlung einer Ziffer, die keine BCD-Ziffer ist, führen.

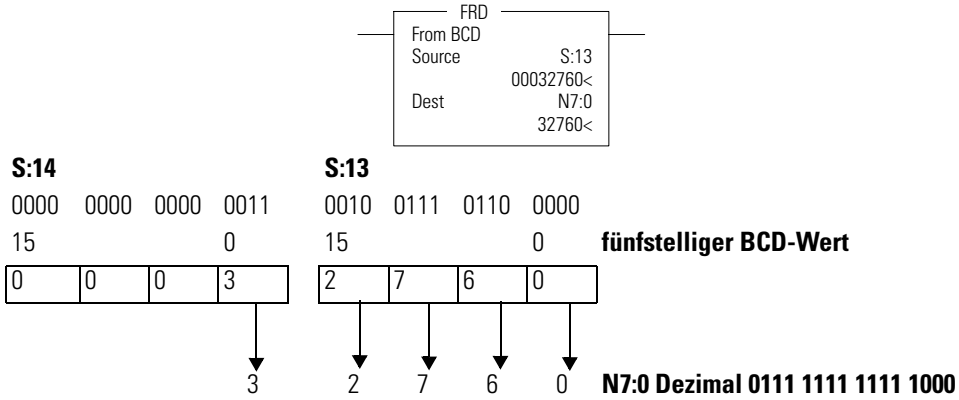



Die beiden dargestellten Strompfade veranlassen die Steuerung vor der Ausführung des FRD-Befehls zu einer Prüfung des Werts I:0, der während zweier aufeinander folgender Abfragen gleich bleiben muss. Auf diese Weise wird verhindert, dass der FRD-Befehl bei Änderung eines Eingangswerts einen Wert umwandelt, der kein BCD-Wert ist.

**HINWEIS** Um BCD-Werte, die größer als 9999 sind, umzu- wandeln, muss als Quellenadresse das Rechenregister (S:13) angegeben werden. Setzen Sie das Bit für geringfügige Fehler (S:5.0), um einen Fehler zu verhindern.

*Beispiel*

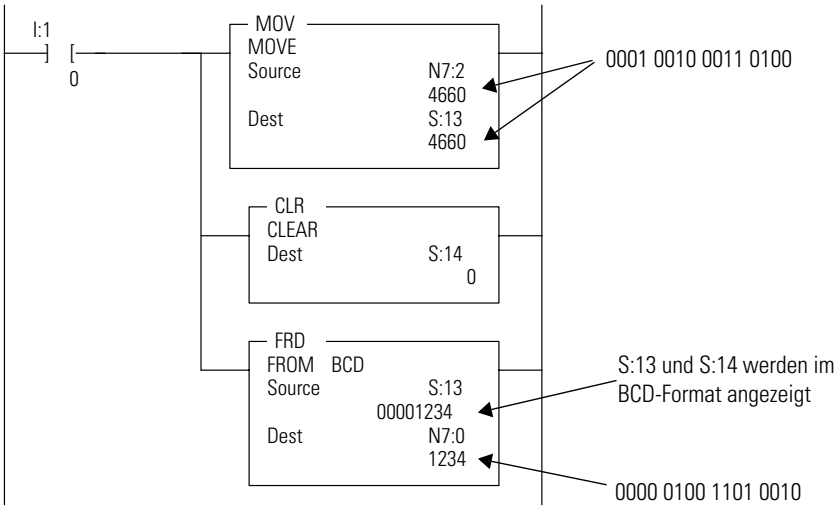
Der BCD-Wert 32 760 im Rechenregister wird umgewandelt und in N7:0 gespeichert. Der maximale BCD-Quellenwert ist 32 767.




Konvertieren Sie BCD-Werte in Ganzzahlwerte, bevor Sie diese im Kontaktplan weiterverarbeiten. Andernfalls werden diese Werte von der Steuerung als ganze Zahlen behandelt, wodurch die Werte selbst verloren gehen können.

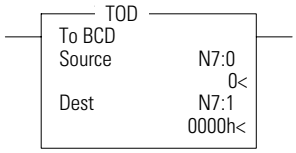
**HINWEIS** Wird das Rechenregister (S:13 und S:14) als Quelle für den FRD-Befehl verwendet und umfasst der BCD-Wert höchstens vier Stellen, ist es erforderlich, Wort S:14 vor Ausführung des FRD-Befehls zu löschen. Wird S:14 nicht gelöscht und wird in diesem Wort ein Wert aus einem anderen mathematischen Befehl gespeichert, der sich an einer anderen Stelle des Programms befindet, wird ein falscher Dezimalwert im Zielwort gespeichert.

Das Löschen von S:14 vor Ausführung des FRD-Befehls wird in der folgenden Abbildung dargestellt:




Wenn der Eingangszustand I:0/1 gesetzt ist (1), wird ein BCD-Wert (der zum Beispiel von einem vierstelligen Daumenradschalter übertragen wurde) von Wort N7:2 in das Rechenregister verschoben. Das Statuswort S:14 wird zurückgesetzt, damit bei der Ausführung des FRD-Befehls keine unerwünschten Daten vorhanden sind.

# TOD – In BCD




Befehlstyp: Ausgang

**Tabelle 11.10 Ausführungszeit des TOD-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 17,2 µs | 0,0 µs |
| MicroLogix 1500 | 14,3 µs | 0,0 µs |

Mit dem TOD-Befehl wird der ganzzahlige Quellenwert in einen BCD-Wert umgewandelt; das Ergebnis wird in das Ziel gestellt.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 11.11 Gültige Adressierungsmodi und Filetypen für TOD-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | Funktionsfiles | | | | | | | | | CS0 - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | • | | • | | |
| Ziel | • | • | • | • | • | • | | | | | | | | | | | | | | | | | | • | • | | • | | (2) |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Siehe unten, „Zieloperand TOD-Befehl".

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

## Zieloperand TOD-Befehl

Bei dem Ziel kann es sich um eine Wortadresse oder das Rechenregister handeln.

Höchstwerte nach Umwandlung in BCD-Format:

- 9999, wenn das Ziel eine Wortadresse ist (nur vierstelliger BCD-Wert möglich).
- 32768, wenn das Ziel das Rechenregister ist (fünfstelliger BCD-Wert möglich, erste vier Stellen in S:13 gespeichert, höhere Stellen in S:14 gespeichert).

Wenn es sich bei dem Ziel um das Rechenregister handelt, muss dieses direkt als S:13 adressiert werden. S:13 ist das einzige Element des Statusfiles, das verwendet werden kann.

## Aktualisierung der mathematischen Status-Bits

**Tabelle 11.12 Mathematische Status-Bits**

| Bit: | | Aktion des Prozessors: |
|---|---|---|
| **S:0/0** | Übertrag | Das Bit wird immer rückgesetzt. |
| **S:0/1** | Überlauf | Das Bit wird gesetzt, wenn das BCD-Ergebnis größer als 9999 ist. Bei einem Überlauf wird außerdem das Flag für geringfügige Störungen gesetzt. |
| **S:0/2** | Null-Bit | Das Bit wird gesetzt, wenn das Ergebnis null ist; andernfalls wird das Bit rückgesetzt. |
| **S:0/3** | Vorzeichen-Bit | Das Bit wird gesetzt, wenn das Quellenwort negativ ist; andernfalls wird das Bit rückgesetzt. |

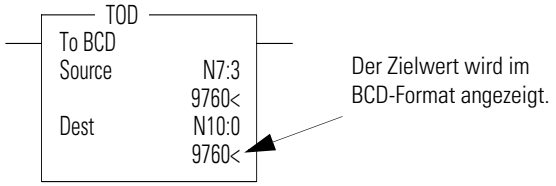
## Änderungen im Rechenregister

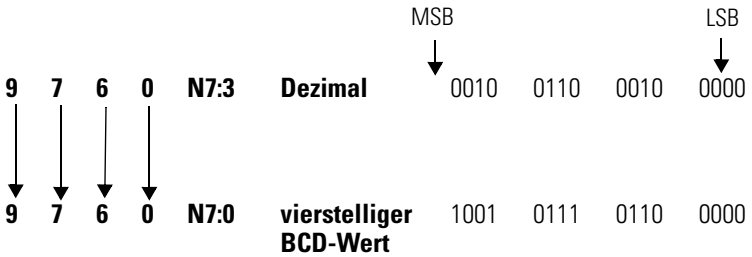
Enthält den fünfstelligen BCD-Wert nach der Konvertierung. Dieser Wert ist bei einem Überlauf gültig.

**HINWEIS** Um BCD-Werte, die größer als 9999 sind, umzu- wandeln, muss als Zieladresse das Rechenregister (S:13) angegeben werden. Setzen Sie das Bit für geringfügige Fehler (S:5/0), um einen Fehler zu verhindern.

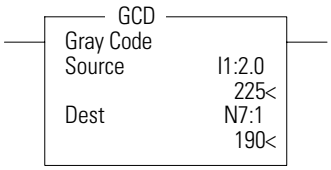
*Beispiel*

Der unter N7:3 gespeicherte Ganzzahlwert 9760 wird in das BCD-Format umgewandelt; der BCD-Wert wird unter N7:0 gespeichert. Der maximal zulässige BCD-Wert ist 9999.



# GCD – Gray-Code




Befehlstyp: Ausgang

**Tabelle 11.13 Ausführungszeit für GCD-Befehle**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 9,5 µs | 0,0 µs |
| MicroLogix 1500 | 8,2 µs | 0,0 µs |

Mit dem GCD-Befehl werden Daten des Gray-Codes (Quelle) in Ganzzahlwerte (Ziel) umgewandelt. Wenn der Eingang des Gray-Codes negativ ist (oberes Bit gesetzt), wird das Ziel auf 32767 gesetzt und das Überlauf-Flag wird gesetzt.

In der folgenden Tabelle werden Adressierungsmodi und Filetypen dargestellt:

**Tabelle 11.14 Gültige Adressierungsmodi und Filetpyen für GCD-Befehl**
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | | | | | | | | | | | | | | | | | | • | • | | • | | |
| Ziel | • | • | | • | • | • | | | | | | | | | | | | | | | | | | | • | • | | • | | |

## Aktualisierung der mathematischen Status-Bits

**Tabelle 11.15 Mathematische Status-Bits**

| Bit: | | Aktion des Prozessors: |
|---|---|---|
| **S:0/0** | Übertrag | Das Bit wird immer rückgesetzt. |
| **S:0/1** | Überlauf | Das Bit wird gesetzt, wenn der Gray-Code-Eingang negativ ist; andernfalls wird es rückgesetzt. |
| **S:0/2** | Null-Bit | Das Bit wird gesetzt, wenn das Ziel null ist; andernfalls wird es rückgesetzt. |
| **S:0/3** | Vorzeichen-Bit | Das Bit wird immer rückgesetzt. |
| **S:5/0** | Überlaufauffang | Das Bit wird gesetzt, wenn das Überlauf-Bit gesetzt wird; andernfalls wird es rückgesetzt. |

# Logikbefehle

Die Logikbefehle führen auf Bitbasis logische Operationen mit einzelnen Worten aus.

| Befehl | Zweck | Seite |
|---|---|---|
| AND - Logisches UND | Ausführung einer logischen UND-Operation | 12-3 |
| OR - Logisches ODER | Ausführung einer Inklusiv-ODER-Operation | 12-4 |
| XOR - Exklusives ODER | Ausführung einer Exklusiv-ODER-Operation | 12-5 |
| NOT - Logisches NICHT | Ausführung einer NICHT-Operation | 12-6 |

## Logikbefehle verwenden

Bei der Verwendung von Logikbefehlen sind folgende Aspekte zu beachten:

- Quelle und Ziel müssen dieselbe Datengröße aufweisen (d. h. beides Worte oder beides Doppelworte).

**WICHTIG** Verwenden Sie den Hochgeschwindigkeits-Zählerakkumulator (HSC.ACC) nicht für den Zieladressenparameter in AND-, OR- und XOR-Befehlen.

- Bei den Quellen A und B kann es sich um Konstanten oder Adressen handeln; allerdings darf nicht für beide Quellen eine Konstante angegeben werden.
- Gültige Bereiche für Konstanten: -32768 bis 32767 (Wort) und -2.147.483.648 bis 2.147.483.647 (Doppelwort).

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 12.1 Gültige Adressierungsmodi und Filetypen für Logikbefehle**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles[1] | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll[2] | Adressierungsmodus[3] | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle A | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Source (Quelle) B[4] | • | • | • | • | • | • | | | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Ziel | • | • | • | • | • | • | | | • | • | | • | | • | • | • | | | • | | | | | | • | • | | • | • | |

(1) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur in MicroLogix 1200- und 1500-BXB-Geräten verwendet.

(2) Der Datenprotokollierungs-Statusfile kann nur durch den MicroLogix 1500 1764-LRP-Prozessor verwendet werden.

(3) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(4) Quelle B wird bei dem NICHT-Befehl nicht verwendet. Der NICHT-Befehl wird nur auf einen Quellenwert angewendet.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

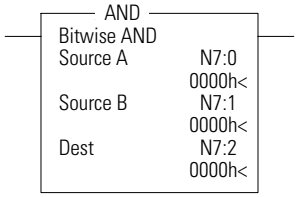
## Aktualisierung der mathematischen Status-Bits

Nach Ausführung eines Logikbefehls werden die mathematischen Status-Bits im Statusfile aktualisiert. Die mathematischen Status-Bits befinden sich in den Bits 0-3 von Wort 0 des Prozessor-Statusfiles (S2).

**Tabelle 12.2 Mathematische Status-Bits**

| Bit: | | Aktion des Prozessors: |
|---|---|---|
| **S:0/0** | Übertrag | Das Bit wird immer rückgesetzt. |
| **S:0/1** | Überlauf | Das Bit wird immer rückgesetzt. |
| **S:0/2** | Null-Bit | Das Bit wird gesetzt, wenn das Ergebnis null ist; andernfalls wird das Bit rückgesetzt. |
| **S:0/3** | Vorzeichen-Bit | Das Bit wird gesetzt, wenn das Ergebnis negativ ist (MSB ist gesetzt); andernfalls wird das Bit rückgesetzt. |

# AND – Logisches UND




Befehlstyp: Ausgang

**Tabelle 12.3 Ausführungszeit des AND-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 2,2 µs | 0,0 µs |
| | Doppelwort | 9,2 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 2,0 µs | 0,0 µs |
| | Doppelwort | 7,9 µs | 0,0 µs |

Verwenden Sie den AND-Befehl, um eine bitweise, logische UND- Operation mit zwei Quellen auszuführen und das Ergebnis in das Ziel zu stellen.

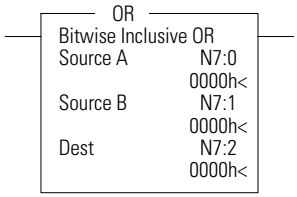
**Tabelle 12.4 Wahrheitstabelle für den AND-Befehl**

| **Ziel = A UND B** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Quelle: A** | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| **Quelle: B** | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| **Ziel:** | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**WICHTIG** Verwenden Sie den Hochgeschwindigkeits-Zählerakkumulator (HSC.ACC) nicht für den Zieladressenparameter in AND-, OR- und XOR-Befehlen.

Weitere Informationen hierzu finden Sie unter „Logikbefehle verwenden“ auf Seite 12-1 und „Aktualisierung der mathematischen Status-Bits“ auf Seite 12-2.

# OR – Logisches ODER




Befehlstyp: Ausgang

**Tabelle 12.5 Ausführungszeit des OR-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 2,2 µs | 0,0 µs |
| | Doppelwort | 9,2 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 2,0 µs | 0,0 µs |
| | Doppelwort | 7,9 µs | 0,0 µs |

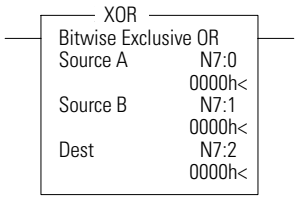
Verwenden Sie den OR-Befehl, um eine bitweise, logische Inklusiv-ODER-Operation mit zwei Quellen auszuführen und das Ergebnis in das Ziel zu stellen.

**Tabelle 12.6 Wahrheitstabelle für den OR-Befehl**

| **Ziel = A ODER B** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Quelle: A** | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| **Quelle: B** | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| **Ziel:** | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |

**WICHTIG** Verwenden Sie den Hochgeschwindigkeits-Zählerakkumulator (HSC.ACC) nicht für den Zieladressenparameter in AND-, OR- und XOR-Befehlen.

# XOR – Exklusives ODER




Befehlstyp: Ausgang

**Tabelle 12.7 Ausführungszeit des XOR-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 3,0 µs | 0,0 µs |
| | Doppelwort | 9,9 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 2,3 µs | 0,0 µs |
| | Doppelwort | 8,9 µs | 0,0 µs |

Verwenden Sie den XOR-Befehl, um eine bitweise, logische Exklusiv-ODER-Operation mit zwei Quellen auszuführen und das Ergebnis in das Ziel zu stellen.

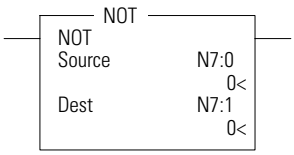
**Tabelle 12.8 Wahrheitstabelle für den XOR-Befehl**

| **Ziel = A ODER B** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Quelle: A** | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| **Quelle: B** | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| **Ziel:** | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |

**WICHTIG** Verwenden Sie den Hochgeschwindigkeits-Zählerakkumulator (HSC.ACC) nicht für den Zieladressenparameter in AND-, OR- und XOR-Befehlen.

Weitere Informationen hierzu finden Sie unter „Logikbefehle verwenden“ auf Seite 12-1 und „Aktualisierung der mathematischen Status-Bits“ auf Seite 12-2.

# NOT – Logisches NICHT




Befehlstyp: Ausgang

**Tabelle 12.9 Ausführungszeit des NOT-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 2,4 µs | 0,0 µs |
| | Doppelwort | 9,2 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 2,4 µs | 0,0 µs |
| | Doppelwort | 8,1 µs | 0,0 µs |

Verwenden Sie den NOT-Befehl, um die Quelle bitweise zu negieren (Einer-Komplements) und das Ergebnis in das Ziel zu stellen.

**Tabelle 12.10 Wahrheitstabelle für den NOT-Befehl**

| **Ziel = A NICHT B** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Quelle:** | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| **Ziel:** | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |

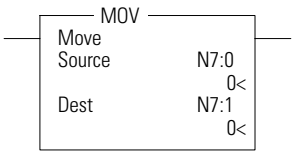
Weitere Informationen hierzu finden Sie unter „Logikbefehle verwenden“ auf Seite 12-1 und „Aktualisierung der mathematischen Status-Bits“ auf Seite 12-2.

# Verschiebebefehle

Die Verschiebebefehle ändern und verschieben Worte.

| Befehl | Zweck | Seite |
|---|---|---|
| MOV - Verschieben | Den Quellenwert zum Ziel verschieben | 13-1 |
| MVM - Maskierte Verschiebung | Daten von einem Quellenstandort an einen ausgewählten Teil des Ziels verschieben | 13-3 |

## MOV – Verschieben




Befehlstyp: Ausgang

**Tabelle 13.1 Ausführungszeit des MOV-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 2,4 µs | 0,0 µs |
| | Doppelwort | 8,3 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 2,3 µs | 0,0 µs |
| | Doppelwort | 6,8 µs | 0,0 µs |

Verwenden Sie den MOV-Befehl, um Daten von der Quelle an das Ziel zu verschieben. Solange der Strompfad wahr bleibt, überträgt der Befehl die Daten bei jeder Abfrage.

### MOV-Befehl verwenden

Bei der Verwendung des MOV-Befehls sind folgende Aspekte zu beachten:

- Die Quelle und das Ziel können unterschiedliche Datengrößen aufweisen. Die Quelle wird bei der Ausführung des Befehls in die Zielgröße umgewandelt. Wenn im Ziel kein ausreichender Speicherplatz für den Quellenwert mit Vorzeichen vorhanden ist, wird der Überlauf wie folgt gehandhabt:
  - Wenn das Bit zur Auswahl der mathematischen Überlauf- funktion nicht gesetzt ist, wird ein gesättigtes Ergebnis im Ziel gespeichert. Wenn die Quelle positiv ist, lautet das Ziel 32767 (Wort). Wenn das Ergebnis negativ ist, lautet das Ziel -32768.
  - Wenn das Bit zur Auswahl der mathematischen Überlauf- funktion gesetzt ist, wird der gekürzte Quellenwert ohne Vorzeichen im Ziel gespeichert.
- Die Quelle kann eine Konstante oder eine Adresse sein.
- Gültige Bereiche für Konstanten: -32768 bis 32767 (Wort) und -2.147.483.648 bis 2.147.483.647 (Doppelwort).

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 13.2 Gültige Adressierungsmodi und Filetypen für MOV-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles[2] | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll[3] | Adressierungsmodus[4] | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F[5] | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | • | • | • | • | • | • | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | | • | • | |
| Ziel | • | • | • | • | • | • | • | • | • | • | | | (6) | (6) | (6) | (6) | | | | | | | | | • | • | | • | • | |

(1) Der ST-File kann nicht für MicroLogix 1500 1764-LSP-Prozessoren der Serie A verwendet werden.

(2) Die DAT-Dateien gelten nur für MicroLogix 1500-Steuerungen. Die PTO- und PWM-Files werden nur in MicroLogix 1200- und 1500-BXB-Geräten verwendet.

(3) Der Datenprotokollierungs-Statusfile kann nur durch den MicroLogix 1500 1764-LRP-Prozessor verwendet werden.

(4) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(5) Der F-File ist nur für MicroLogix 1200- und 1500-Steuerungen der Serie C und höher gültig.

(6) Einige Elemente sind beschreibbar. Ausführliche Informationen finden Sie im Funktionsfile.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

## Aktualisierung der mathematischen Status-Bits

Nach Ausführung eines MOV-Befehls werden die mathematischen Status-Bits im Statusfile aktualisiert. Die arithmetischen Status-Bits befinden sich in den Bits 0 bis 3 von Wort 0 des Prozessor-Statusfiles (S2).

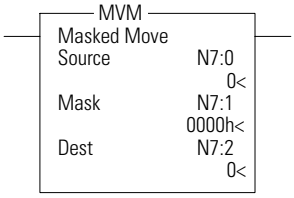
**Tabelle 13.3 Mathematische Status-Bits**

| Bit: | | Aktion des Prozessors: |
|---|---|---|
| **S:0/0** | Übertrag | Das Bit wird immer rückgesetzt. |
| **S:0/1** | Überlauf | Das Bit wird gesetzt, wenn ein Überlauf, unendlich oder NAN (not a number; keine Nummer), entdeckt wurde; andernfalls wird es rückgesetzt. |
| **S:0/2** | Null-Bit | Das Bit wird gesetzt, wenn das Ergebnis null ist; andernfalls wird das Bit rückgesetzt. |
| **S:0/3** | Vorzeichen-Bit | Das Bit wird gesetzt, wenn das Ergebnis negativ ist (MSB ist gesetzt); andernfalls wird das Bit rückgesetzt. |
| **S:5/0** | Erkennungs-Bit für den mathematischen Überlauf[1] | Im Erkennungs-Bit für den mathematischen Überlauf wird ein geringfügiger Fehler generiert, wenn das Überlauf-Bit gesetzt ist; andernfalls behält das Bit seinen letzten Status bei. |

(1) Steuer-Bit

**HINWEIS** Wenn ein Datenwort ohne Änderung der mathe- matischen Flags verschoben werden soll, sollte anstatt des MOV-Befehls der Kopierbefehl (COP) mit einer Länge von einem Wort verwendet werden.

# MVM – Maskierte Verschiebung




Befehlstyp: Ausgang

**Tabelle 13.4 Ausführungszeit des MVM-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 7,8 µs | 0,0 µs |
| | Doppelwort | 11,8 µs | 0,0 µs |
| MicroLogix 1500 | Wort | 7,2 µs | 0,0 µs |
| | Doppelwort | 10,0 µs | 0,0 µs |

Verwenden Sie den MVM-Befehl, um Daten von der Quelle an das Ziel zu verschieben und dabei Teile des Ziels auszumaskieren. Funktionsweise der Maske:

**Tabelle 13.5 Maskenfunktion bei MVM-Befehl**

| Quellen-Bit | Masken-Bit | Ziel-Bit |
|---|---|---|
| 1 | 0 | letzter Status |
| 0 | 0 | letzter Status |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

Um Daten auszumaskieren (Verschiebung sperren), setzen Sie die entsprechenden Masken-Bits auf Null; um Daten zu verschieben, setzen Sie das entsprechende Masken-Bit auf Eins. Die Maske kann entweder eine Konstante oder durch Zuordnen einer direkten Adresse auch variabel sein. Bits der Zieladresse, die einer 0 in der Maske entsprechen, werden nicht verändert.

## MVM-Befehl verwenden

Bei der Verwendung des MVM-Befehls sind folgende Aspekte zu beachten:

- Quelle, Maske und Ziel müssen dieselbe Datengröße aufweisen (d. h. nur Worte oder nur Doppelworte).

  Um Daten auszumaskieren (Verschiebung sperren), setzen Sie das Masken-Bit auf Null; um Daten zu verschieben, setzen Sie das Masken-Bit auf Eins. Die Maske kann entweder eine Konstante oder durch Zuordnen einer direkten Adresse auch variabel sein.

  **HINWEIS** Bits an der Zieladresse, die den Null-Werten der Maske entsprechen, werden nicht geändert (siehe grau unterlegte Bereiche der folgenden Tabelle).

**Tabelle 13.6 Beispiel für Maske (Wortadressierungsebene)**

| Wort | Hexadezimalwert | Binärwert | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Wert an Zieladresse vor Verschieben | FFFF | 1 | 1 | 1 | 1 | **1** | **1** | **1** | **1** | 1 | 1 | 1 | 1 | **1** | **1** | **1** | **1** |
| Quellenwert | 5555 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| Maske | F0F0 | 1 | 1 | 1 | 1 | **0** | **0** | **0** | **0** | 1 | 1 | 1 | 1 | **0** | **0** | **0** | **0** |
| Wert an Zieladresse nach Verschieben | 5F5F | 0 | 1 | 0 | 1 | **1** | **1** | **1** | **1** | 0 | 1 | 0 | 1 | **1** | **1** | **1** | **1** |

- Gültige Bereiche für die Verwendung von Konstanten in der Maske: -32768 bis 32767 (Wort) und -2.147.483.648 bis 2.147.483.647 (Doppelwort). Die Maske wird als hexadezimaler Wert ohne Vorzeichen im Bereich von 0000 0000 bis FFFF FFFF dargestellt.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 13.7 Gültige Adressierungsmodi und Filetypen für den MVM-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus[2] | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | • | • | | | | | | | | | | | | | | | | • | • | | | • | |
| Maske | • | • | | • | • | • | | • | • | | | | | | | | | | | | | | | • | • | • | | | • | |
| Ziel | • | • | | • | • | • | | • | • | | | | | | | | | | | | | | | | • | • | | | • | |

(1) Der ST-File kann nicht für MicroLogix 1500 1764-LSP-Prozessoren der Serie A verwendet werden.

(2) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

## Aktualisierung der mathematischen Status-Bits

Nach Ausführung eines MVM-Befehls werden die mathematischen Status-Bits im Statusfile aktualisiert. Die mathematischen Status-Bits befinden sich in den Bits 0-3 von Wort 0 des Prozessor-Statusfiles (S2).

**Tabelle 13.8 Mathematische Status-Bits**

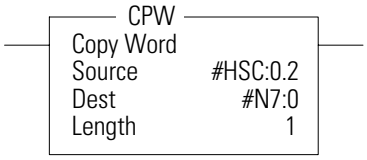
| Bit: | | Aktion des Prozessors: |
|---|---|---|
| **S:0/0** | Übertrag | Das Bit wird immer rückgesetzt. |
| **S:0/1** | Überlauf | Das Bit wird immer rückgesetzt. |
| **S:0/2** | Null-Bit | Das Bit wird gesetzt, wenn das Ziel null ist; andernfalls wird das Bit rückgesetzt. |
| **S:0/3** | Vorzeichen-Bit | Das Bit wird gesetzt, wenn das MSB des Ziels gesetzt ist; andernfalls wird das Bit rückgesetzt. |

# Filebefehle

Die Filebefehle führen Operationen an Filedaten aus.

| Befehl | Zweck | Seite |
|---|---|---|
| CPW - Wort kopieren | Wörter aus Daten von einem Standort an einen anderen Standort kopieren | 14-4 |
| COP - File kopieren | Daten von einem Filestandort an einen anderen kopieren | 14-4 |
| FLL - Filefüllung | File mit einer Programmkonstante oder einem Wert aus einer Elementadresse laden | 14-5 |
| BSL - Bit nach links verschieben | Daten bitweise in ein Feld laden und aus dem Feld entladen | 14-6 |
| BSR - Bit nach rechts verschieben | | 14-8 |
| FFL - FIFO laden | Daten in einen File laden und in derselben Reihenfolge entladen (FIFO) | 14-10 |
| FFU - FIFO entladen | | 14-13 |
| LFL - LIFO laden | Daten in einen File laden und in der umgekehrten Reihenfolge entladen (LIFO) | 14-15 |
| LFU - LIFO entladen | | 14-17 |
| SWP - Byte-Tausch (*Nur MicroLogix 1200- und 1500-Steuerungen der Serie B und höher*) | Vertauschen des niederwertigen Bytes mit dem höherwertigen Byte in einer angegebenen Anzahl von Wörtern | 14-19 |

# CPW – Wort kopieren




Befehlstyp: Ausgang

**Tabelle 14.1 Ausführungszeit des CPW-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| Nur MicroLogix 1200 Serie C und höher | 18,3 µs + 0,8 µs/Wort | 0,0 µs |
| nur MicroLogix 1500 Serie C und höher | 15,8 µs + 0,7 µs/Wort | 0,0 µs |

Mithilfe des CWP-Befehls können Sie Wörter in aufsteigender Reihenfolge von einem Standort (Quelle) zu einem anderen (Ziel) kopieren. Dieser Befehl ähnelt dem COP-Befehl (Datei kopieren); mit dem CPW-Befehl ist es jedoch möglich, unterschiedliche Quell- und Zielparameter zu haben. Beispiele sind:

- Ganzzahl nach Doppelwort
- Doppelwort nach Fließkomma
- Doppelwort nach Ganzzahl
- Ganzzahl nach PTO-Funktionsfile

Bei der Verwendung des CPW-Befehls sind folgende Einschränkungen zu beachten:

- Die Länge der übermittelten Daten darf nicht mehr als 128 Wörter betragen.
- Funktionsfiles können als Quelle oder Ziel verwendet werden, jedoch nicht als beides.
- Wenn Sie entweder auf einen PLS-File oder einen Funktionsfile verweisen, muss die Adressierung auf das Unterelement-Level angepasst sein.
- Sie können auf ein Unterelement aus Funktionsfile-Bits verweisen, das aus einer Kombination aus Nur Lesen-Bits und Lesen/Schreiben-Bits besteht.
- Sie können nicht direkt auf das obere Wort eines Doppelwortes als Operand in dem CPW-Befehl verweisen.
- Ein schwerer Fehler (003F) tritt auf, wenn die Ausführung des Befehls den Datentabellen-Platz übersteigt.
- Ein schwerer Fehler (0044) tritt auf, wenn ein Schreibversuch in den RTC-Funktionsfile fehlschlägt. Dies tritt nur auf, wenn versucht wird, ungültige Daten in den RTC-Funktionsfile zu schreiben. Beispiele für ungültige Daten sind: Das Setzen des Wochentages auf Null oder das Setzen des Datums auf den 30. Februar.

In der folgenden Tabelle werden Adressierungsmodi und Filetypen dargestellt:

**Tabelle 14.2 Gültige Adressierungsmodi und Filetypen für CPW-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt„Befehlsbeschreibungen verwenden" auf Seite 4-2 .*

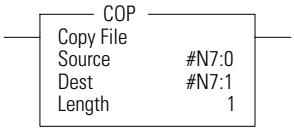
| Parameter | Datenfiles | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus[1] | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F[2] | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | | • | • | | • | | • | • | • | • | • | • | • | • | | • | | | | | • | • | | | | • |
| Ziel | • | • | | • | | • | • | | • | | • | • | • | • | • | • | | | | | | | | | • | • | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | • | | | | | | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Der F-File ist nur für MicroLogix 1200- und 1500-Steuerungen der Serie C und höher gültig.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# COP – File kopieren




Befehlstyp: Ausgang

**Tabelle 14.3 Ausführungszeit des COP-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 19,08 µs + 0,8 µs/Wort | 0,0 µs |
| MicroLogix 1500 | 15,9 µs + 0,67 µs/Wort | 0,0 µs |

Mit dem COP-Befehl werden Datenblöcke von einem Speicherplatz an einen anderen kopiert.

**Tabelle 14.4 Gültige Adressierungsmodi und Filetpyen für COP-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus[2] | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | • | • | • | | | | | | | | | | | | | | | | • | • | | | | • |
| Ziel | • | • | | • | • | • | • | • | • | | | | | | | | | | | | | | | | • | • | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | | | |

(1) Der ST-File kann nicht für MicroLogix 1500 1764-LSP-Prozessoren der Serie A verwendet werden.

(2) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

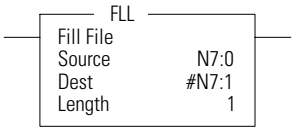
**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

Die Quell- und Ziel-Filetypen müssen mit Ausnahme von Bit (B) und Ganzzahl (N) identisch sein; sie sind austauschbar. Die maximal zu kopierende Blocklänge wird durch die Adresse festgelegt (siehe nachfolgende Tabelle).

**Tabelle 14.5 Maximale Blockgröße bei COP-Befehl**

| Datentyp Quelle/Ziel | Bereich der Operandenlänge |
|---|---|
| 1-Wort-Elemente (d. h. Wort) | 1 bis 128 |
| 2-Wort-Elemente (d. h. Doppelwort) | 1 bis 64 |
| 3-Wort-Elemente (d. h. Zähler) | 1 bis 42 |
| 42-Wort-Elemente (d. h. Zeichenkette) | 1 bis 3 |

# FLL – Filefüllung

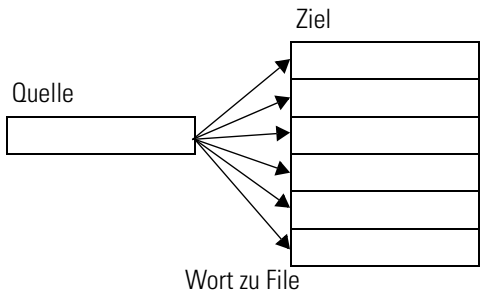



Befehlstyp: Ausgang

**Tabelle 14.6 Ausführungszeit des FLL-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 14 + 0,6 µs/Wort | 0,0 µs |
| | Doppelwort | 15 + 1,2 µs/Doppelwort | 0,0 µs |
| MicroLogix 1500 | Wort | 12,1 + 0,43 µs/Wort | 0,0 µs |
| | Doppelwort | 12,3 + 0,8 µs/Doppelwort | 0,0 µs |

Mit dem FLL-Befehl werden Elemente eines Files mit einer Konstante oder einem Adressdatenwert einer bestimmten Länge geladen. Die folgende Abbildung zeigt, wie Filebefehlsdaten verarbeitet werden. Der Befehl füllt die Worte eines Files mit einem Quellenwert. Er setzt keine Status-Bits. Wenn Sie ein Freigabe-Bit benötigen, so program- mieren Sie einen parallelen Ausgang, der eine Speicheradresse verwendet.




Dieser Befehl verwendet folgende Operanden:

- Quelle - Der Quellenoperand ist die Adresse des Wertes oder der Konstante, der/die zum Füllen des Ziels verwendet wird. Die Quellenwerte müssen in dem Bereich -32768 bis 32767 (Wort), -2 147 483 648 bis 2 147 483 647 (Doppelwort) oder einem IEEE-754-32-Bit-Wert liegen.

**HINWEIS** In Zeitwerkfiles (T), Zählerfiles (C) oder Steuerfiles (R) können keine Konstanten als Quelle verwendet werden.

- Ziel - Die beginnende Zieladresse, in die die Daten geschrieben werden.
- Länge - Der Längenoperand enthält die Anzahl der Elemente. Die Länge kann zwischen 1 und 128 (Wort), 1 und 64 (Doppelwort) oder 1 und 42 (3-Wort-Elemente wie Zählwerk) betragen.

**HINWEIS** Die Operanden für Quelle und Ziel müssen identisch sein, sofern es sich nicht um Bit (B) oder Ganzzahl (N) handelt.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 14.7 Gültige Adressierungsmodi und Filetypen für FLL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

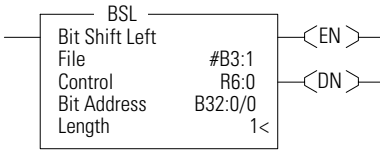
| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F(2) | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | • | | • | | | | | | | | | | | | | | | • | • | • | | • | • | • |
| Ziel | • | • | | • | • | • | • | | • | | | | | | | | | | | | | | | | • | • | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Der F-File ist nur für MicroLogix 1200- und 1500-Steuerungen der Serie C und höher gültig.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DATI-, TPI-, CS-, IOS- und DLS-Files.

# BSL – Bit nach links verschieben

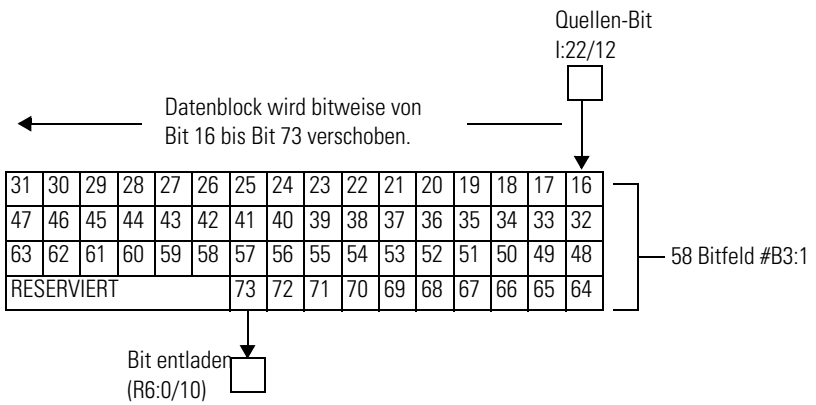



Befehlstyp: Ausgang

**Tabelle 14.8 Ausführungszeit des BSL-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 32 µs + 1,3 µs/Wort | 1,3 µs |
| MicroLogix 1500 | 26,1 µs + 1,06 µs/Wort | 1,4 µs |

Mit dem BSL-Befehl werden Daten bei einem Strompfadübergang von unwahr nach wahr bitweise in ein Datenfeld geladen. Die Daten werden dann innerhalb des Feldes nach links verschoben und anschließend bitweise aus dem Feld entladen. Die folgende Abbildung zeigt die Funktionsweise des BSL-Befehls.



Wenn Sie in einem Abfragezyklus mehrere Bits verschieben möchten, muss in der Anwendung mit den Befehlen JMP, LBL und CTU eine Schleife erstellt werden.

Dieser Befehl verwendet folgende Operanden:

- File - Der Fileoperand ist die Adresse des Bitfelds, das geändert werden soll.
- Steuerung - Der Steuerungsoperand ist die Adresse des Steuer- elements des BSL-Befehls. Das Steuerelement besteht aus drei Worten:

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort 0** | EN(1) | -- | DN(2) | -- | ER(3) | UL(4) | nicht belegt | | | | | | | | | |
| **Wort 1** | Größe eines Bitfelds (Anzahl Bits). | | | | | | | | | | | | | | | |
| **Wort 2** | nicht belegt | | | | | | | | | | | | | | | |

(1) EN - Das Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde.

(2) DN - Fertig-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass das Bitfeld um eine Position verschoben wurde.

(3) ER - Fehler-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass ein Fehler festgestellt wurde, z. B. Eingabe einer negativen Zahl für die Länge oder den Quellenoperanden.

(4) UL - Entlade-Bit; der Ausgang des Befehls. Verwenden Sie das Entlade-Bit nicht, wenn das Fehler-Bit gesetzt ist.

- Bitadresse - Die Quelle entspricht der Adresse des Bits, das an die erste (niedrigste) Stelle im Bitfeld übertragen werden soll.
- Länge - Der Längenoperand enthält die Länge des Bitfelds in Bit. Der gültige Datenbereich für die Länge liegt zwischen 0 und 2048.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 14.9 Gültige Adressierungsmodi und Filetpyen für BSL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

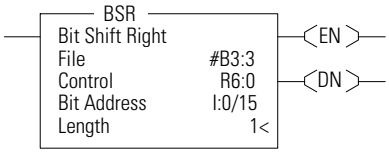
| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| File | • | • | | • | | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Steuerung | | | | | (2) | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Quelle | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | | | • | • | • | | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Nur Steuerfile. Gilt nicht für Zeitwerke und Zähler.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DATI-, TPI-, CS-, IOS- und DLS-Files.

# BSR – Bit nach rechts verschieben



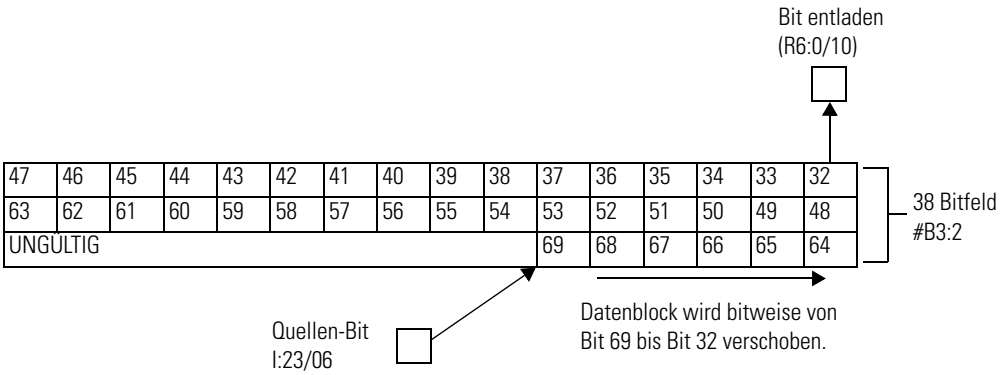

Befehlstyp: Ausgang

**Tabelle 14.10 Ausführungszeit des BSR-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | **wahr** | **unwahr** |
| MicroLogix 1200 | 32 µs + 1,3 µs/Wort | 1,3 µs |
| MicroLogix 1500 | 26,1 µs + 1,07 µs/Wort | 1,4 µs |

Wenn Sie in einem Abfragezyklus mehrere Bits verschieben möchten, muss in der Anwendung mit den Befehlen JMP, LBL und CTU eine Schleife erstellt werden.

Mit dem BSR-Befehl werden Daten bei einem Strompfadübergang von unwahr nach wahr bitweise in ein Datenfeld geladen. Die Daten werden dann innerhalb des Feldes nach rechts verschoben und anschließend bitweise aus dem Feld entladen. Die folgende Abbildung zeigt die Funktionsweise des BSR-Befehls.




Dieser Befehl verwendet folgende Operanden:

- File - Der Fileoperand ist die Adresse des Bitfelds, das geändert werden soll.
- Steuerung - Der Steuerungsoperand ist die Adresse des Steuerelements des BSR-Befehls. Das Steuerelement besteht aus drei Worten:

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort 0** | EN(1) | -- | DN(2) | -- | ER(3) | UL(4) | nicht belegt | | | | | | | | | |
| **Wort 1** | Größe eines Bitfelds (Anzahl Bits). | | | | | | | | | | | | | | | |
| **Wort 2** | nicht belegt | | | | | | | | | | | | | | | |

(1) EN - Das Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde.

(2) DN - Fertig-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass das Bitfeld um eine Position verschoben wurde.

(3) ER - Fehler-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass ein Fehler festgestellt wurde, z. B. Eingabe einer negativen Zahl für die Länge oder den Quellenoperanden.

(4) UL - Entlade-Bit; der Ausgang des Befehls. Verwenden Sie das Entlade-Bit nicht, wenn das Fehler-Bit gesetzt ist.

- Bitadresse - Die Quelle entspricht der Adresse des Bits, das an die letzte (höchste) Stelle im Bitfeld übertragen werden soll.
- Länge - Der Längenoperand enthält die Länge des Bitfelds in Bit. Der Datenbereich für die Länge liegt zwischen 0 und 2048.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 14.11 Gültige Adressierungsmodi und Filetpyen für BSR-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

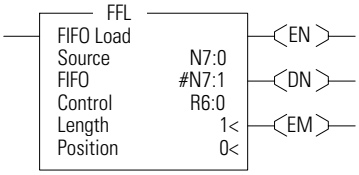
| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | L | ST | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| File | • | • | | • | | • | | • | | | | | | | | | | | | | | | | | • | • | | • | • | |
| Steuerung | | | | | (2) | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Quelle | • | • | | • | • | • | | • | | | | | | | | | | | | | | | | | • | • | • | | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Nur Steuerfile. Gilt nicht für Zeitwerke und Zähler.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# FFL – FIFO laden

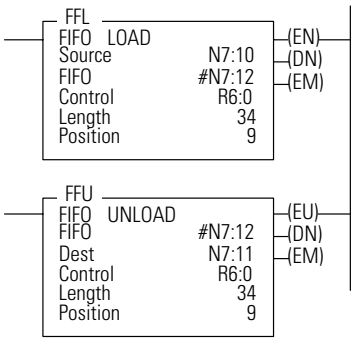



Befehlstyp: Ausgang

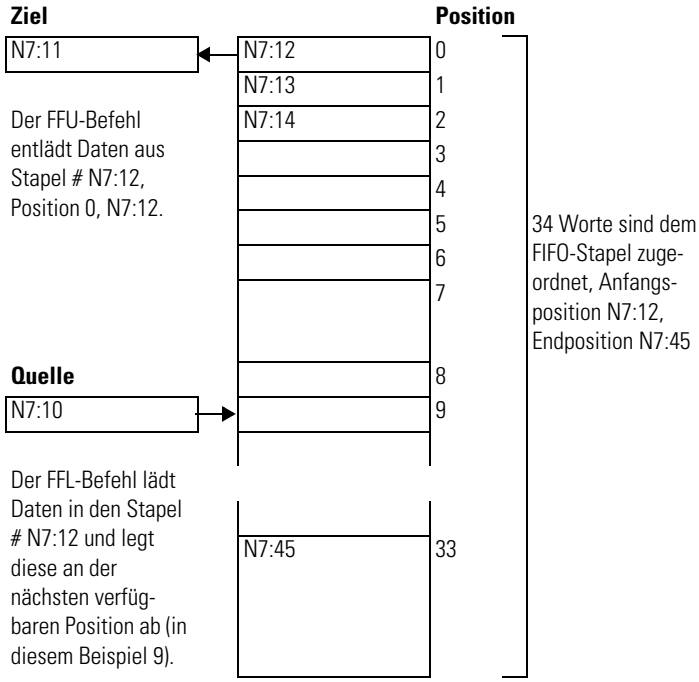
**Tabelle 14.12 Ausführungszeit des FFL-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 11,3 µs | 11,1 µs |
| | Doppelwort | 11,7 µs | 11,2 µs |
| MicroLogix 1500 | Wort | 10,0 µs | 9,8 µs |
| | Doppelwort | 10,9 µs | 9,7 µs |

Mit dem FFL-Befehl werden bei einem Strompfadübergang von unwahr nach wahr Worte oder Doppelworte in einen vom Anwender erstellten File, den so genannten FIFO-Stapel, geladen. Zu jedem FFL-Befehl ist als Gegenstück ein FFU-Befehl (FIFO entladen) vor- handen, mit dem Elemente aus dem FIFO-Stapel entladen werden. In dem nachfolgend dargestellten FFL/ FFU-Befehlspaar wurden Befehls- parameter programmiert.




**Befehlspaar FFL und FFU**




**Laden und Entladen des Stapels # N7:12**

Dieser Befehl verwendet folgende Operanden:

- Quelle - Der Quellenoperand ist eine Konstante oder eine Adresse mit dem Wert, der an die momentan verfügbare Position in dem FIFO-Stapel gestellt werden soll. Die Adressierungsebene der Quelle muss mit der des FIFO-Stapels übereinstimmen. Wenn es sich bei FIFO um einen Wort-File handelt, muss die Quelle ein Wort-Wert oder eine Wort-Konstante sein. Wenn es sich bei FIFO um einen Doppelwort-File handelt, muss die Quelle ein Doppelwort-Wert oder eine Doppelwort-Konstante sein. Die Quellenwerte müssen in dem Bereich -32768 bis 32767 (Wort) oder -2.147.483.648 bis 2.147.483.647 (Doppelwort) liegen.
- FIFO - Der FIFO-Operand ist die Anfangsadresse des Stapels.
- Steuerung - Dies ist die Adresse des Steuerfiles. Dieses Element enthält die Status-Bits, die Stapellänge und den Positionswert. Das Steuerelement besteht aus drei Worten:

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort 0** | EN(1) | -- | DN(2) | EM(3) | nicht belegt | | | | | | | | | | | |
| **Wort 1** | Länge - die maximale Anzahl Worte oder Doppelworte in dem Stapel | | | | | | | | | | | | | | | |
| **Wort 2** | Position - der nächste verfügbare Standort, an den Daten geladen werden können | | | | | | | | | | | | | | | |

(1) EN - Das Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde.

(2) DN - Fertig-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass der Stapel voll ist.

(3) EM - Leer-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass der Stapel leer ist.

- Länge - Der Längenoperand enthält die Anzahl Elemente in dem FIFO-Stapel, in den der Wert oder die Konstante aus der Quelle geladen werden soll. Die Länge des Stapels kann zwischen 1 und 128 (Wort) oder 1 und 64 (Doppelwort) betragen. Die Position wird nach jedem Ladevorgang erhöht.
- Position - Dies ist der aktuelle Standort, auf den in dem FIFO- Stapel verwiesen wird. Die Position bezeichnet den nächsten verfügbaren Standort im Stapel, an dem der Wert oder die Konstante aus der Quelle gespeichert wird. Die Position ist eine Komponente des Steuerregisters. Die Position kann zwischen 0 und 127 (Wort) oder 0 und 63 (Doppelwort) betragen.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 14.13 Gültige Adressierungsmodi und Filetpyen für FFL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

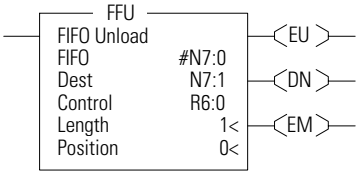
| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| FIFO | • | • | | • | | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Steuerung | | | | | (2) | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Position | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Nur Steuerfile. Gilt nicht für Zeitwerke und Zähler.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# FFU – FIFO entladen

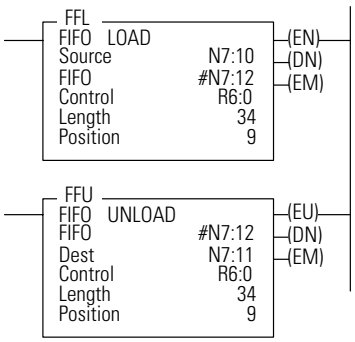



Befehlstyp: Ausgang

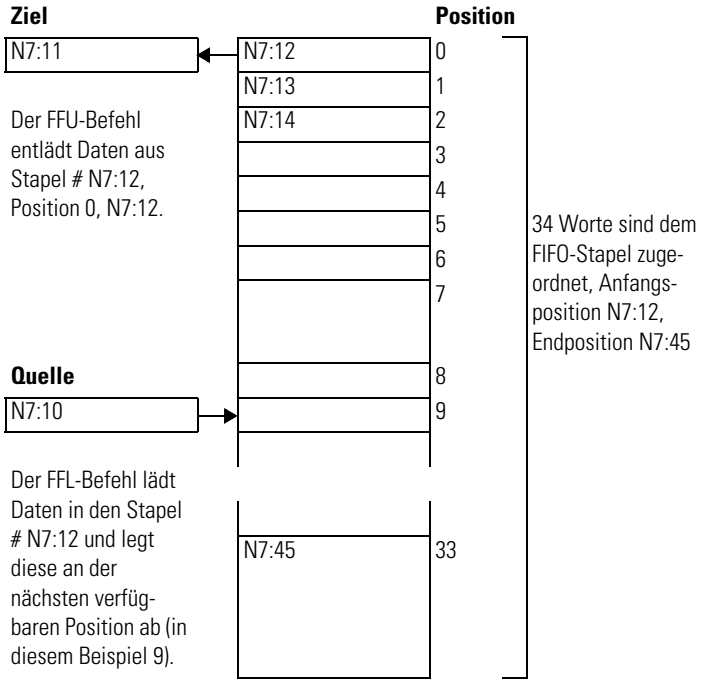
**Tabelle 14.14 Ausführungszeit des FFU-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | **wahr** | **unwahr** |
| MicroLogix 1200 | Wort | 33 µs + 0,8 µs/Wort | 10,4 µs |
| | Doppelwort | 36 µs + 1,5 µs/Doppelwort | 10,4 µs |
| MicroLogix 1500 | Wort | 27,7 µs + 0,65 µs/Wort | 9,7 µs |
| | Doppelwort | 29,4 µs + 1,25 µs/Doppelwort | 9,7 µs |

Mit dem FFU-Befehl werden bei einem Strompfadübergang von unwahr nach wahr Worte oder Doppelworte aus einem vom Anwender erstellten File, dem so genannten FIFO-Stapel, entladen. Die Daten werden entsprechend dem FIFO-Prinzip entladen. Nach Abschluss des Entladevorgangs werden die Daten in dem Stapel um ein Element nach oben verschoben, und das letzte Element wird gelöscht. In dem nachfolgend dargestellten FFL/FFU-Befehlspaar wurden Befehlsparameter programmiert.




**Befehlspaar FFL und FFU**




**Laden und Entladen des Stapels # N7:12**

Dieser Befehl verwendet folgende Operanden:

- FIFO - Der FIFO-Operand ist die Anfangsadresse des Stapels.
- Ziel - Der Zieloperand ist eine Wort- oder Doppelwortadresse, an der der Wert nach dem Entladen aus dem FIFO-Stapel gespeichert wird. Der FFU-Befehl entlädt diesen Wert aus dem ersten Standort des FIFO-Stapels und speichert ihn unter der Zieladresse. Die Adressierungsebene des Ziels muss mit der des FIFO-Stapels übereinstimmen. Wenn es sich bei FIFO um einen Wort-File handelt, muss das Ziel ein Wort-File sein. Wenn es sich bei FIFO um einen Doppelwort-File handelt, muss das Ziel ein Doppelwort-File sein.
- Steuerung - Dies ist die Adresse des Steuerfiles. Dieses Element enthält die Status-Bits, die Stapellänge und den Positionswert. Das Steuerelement besteht aus drei Worten:

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort 0** | -- | EU(1) | DN(2) | EM(3) | nicht belegt | | | | | | | | | | | |
| **Wort 1** | Länge - die maximale Anzahl Worte oder Doppelworte in dem Stapel | | | | | | | | | | | | | | | |
| **Wort 2** | Position - der nächste verfügbare Standort, an den Daten entladen werden können | | | | | | | | | | | | | | | |

(1) EU - Das FFU-Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde.

(2) DN - Fertig-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass der Stapel voll ist.

(3) EM - Leer-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass der Stapel leer ist.

- Länge - Der Längenoperand enthält die Anzahl der Elemente in dem FIFO-Stapel. Die Länge des Stapels kann zwischen 1 und 128 (Wort) oder 1 und 64 (Doppelwort) betragen.
- Position - Die Position ist eine Komponente des Steuerregisters. Die Position kann zwischen 0 und 127 (Wort) oder 0 und 63 (Doppelwort) betragen. Die Position wird nach jedem Lade- vorgang reduziert. Daten werden bei Position null entladen.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 14.15 Gültige Adressierungsmodi und Filetpyen für FFU-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

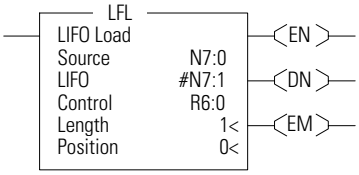
| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| FIFO | • | • | | • | | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Ziel | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Steuerung | | | | | (2) | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Position | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Nur Steuerfile. Gilt nicht für Zeitwerke und Zähler.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# LFL – LIFO laden

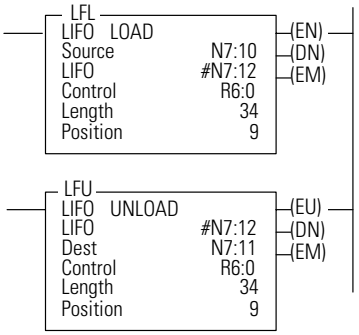



Befehlstyp: Ausgang

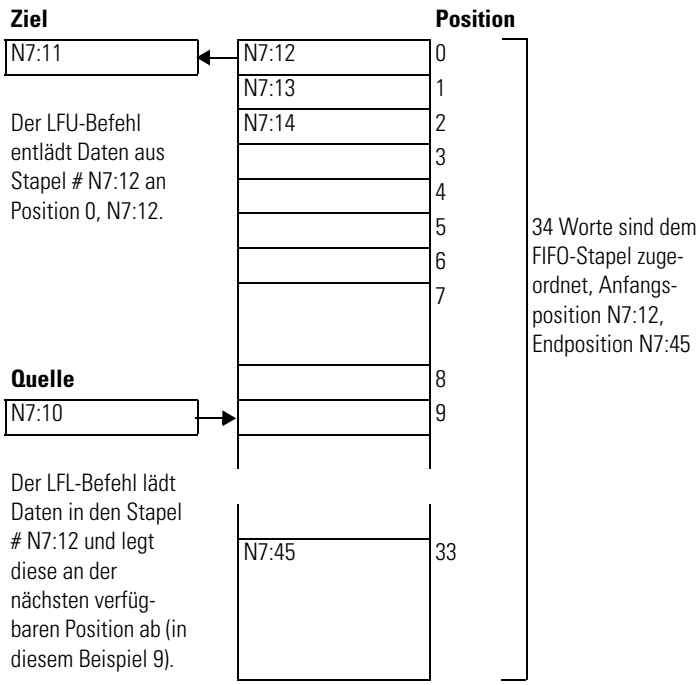
**Tabelle 14.16 Ausführungszeit des LFL-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 25,5 µs | 10,4 µs |
| | Doppelwort | 31,6 µs | 10,4 µs |
| MicroLogix 1500 | Wort | 22,2 µs | 9,7 µs |
| | Doppelwort | 27,4 µs | 9,7 µs |

Mit dem LFL-Befehl werden bei einem Strompfadübergang von unwahr nach wahr Worte oder Doppelworte in einen vom Anwender erstellten File, den so genannten LIFO-Stapel, geladen. Zu jedem LFL-Befehl ist als Gegenstück ein LFU-Befehl (LIFO entladen) vor- handen, mit dem Elemente aus dem LIFO-Stapel entladen werden. Im nachfolgend dargestellten LFL/ LFU-Befehlspaar wurden Befehls- parameter programmiert.




**Befehlspaar LFL und LFU**




**Laden und Entladen des Stapels # N7:12**

Dieser Befehl verwendet folgende Operanden:

- Quelle - Der Quellenoperand ist eine Konstante oder eine Adresse mit dem Wert, der an die momentan verfügbare Position in dem LIFO-Stapel gestellt werden soll. Die Datengröße der Quelle muss mit der des LIFO-Stapels übereinstimmen. Wenn es sich bei LIFO um einen Wort-File handelt, muss die Quelle ein Wort-Wert oder eine Wort-Konstante sein. Wenn es sich bei LIFO um einen Doppelwort-File handelt, muss die Quelle ein Doppelwort-Wert oder eine Doppelwort-Konstante sein. Die Quellenwerte müssen in dem Bereich -32768 bis 32767 (Wort) oder -2.147.483.648 bis 2.147.483.647 (Doppelwort) liegen.

- LIFO - Der LIFO-Operand ist die Anfangsadresse des Stapels.
- Steuerung - Dies ist die Adresse des Steuerfiles. Dieses Element enthält die Status-Bits, die Stapellänge und den Positionswert. Das Steuerelement besteht aus drei Worten:

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort 0** | EN(1) | -- | DN(2) | EM(3) | nicht belegt | | | | | | | | | | | |
| **Wort 1** | Länge - die maximale Anzahl Worte oder Doppelworte in dem Stapel | | | | | | | | | | | | | | | |
| **Wort 2** | Position - der nächste verfügbare Standort, an den Daten geladen werden können | | | | | | | | | | | | | | | |

(1) EN - Das Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde.

(2) DN - Fertig-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass der Stapel voll ist.

(3) EM - Leer-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass der Stapel leer ist.

- Länge - Der Längenoperand enthält die Anzahl Elemente in dem FIFO-Stapel, in den der Wert oder die Konstante aus der Quelle geladen werden soll. Die Länge des Stapels kann zwischen 1 und 128 (Wort) oder 1 und 64 (Doppelwort) betragen. Die Position wird nach jedem Ladevorgang erhöht.
- Position - Dies ist der aktuelle Standort, auf den in dem LIFO-Stapel verwiesen wird. Die Position bezeichnet den nächsten verfügbaren Standort im Stapel, an dem der Wert oder die Konstante aus der Quelle gespeichert wird. Die Position ist eine Komponente des Steuerregisters. Die Position kann zwischen 0 und 127 (Wort) oder 0 und 63 (Doppelwort) betragen.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 14.17 Gültige Adressierungsmodi und Filetpyen für LFL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

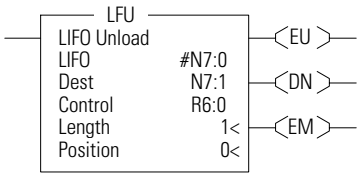
| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| LIFO | • | • | | • | | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Steuerung | | | | | (2) | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Position | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Nur Steuerfile. Gilt nicht für Zeitwerke und Zähler.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# LFU – LIFO entladen

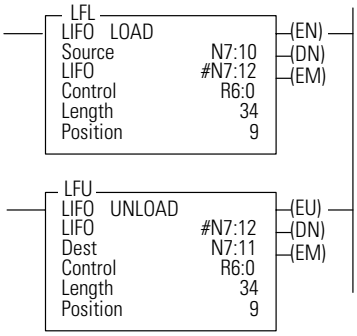



Befehlstyp: Ausgang

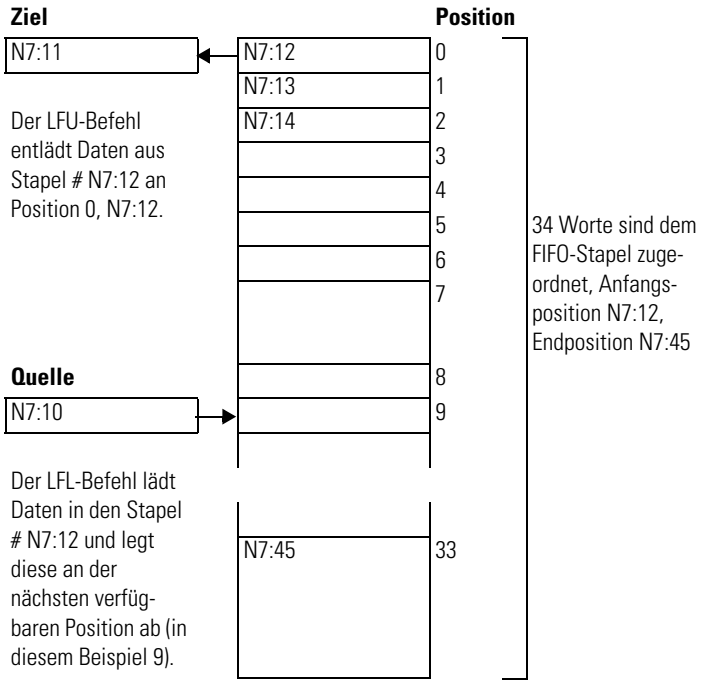
**Tabelle 14.18 Ausführungszeit des LFU-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 29,1 µs | 10,4 µs |
| | Doppelwort | 31,6 µs | 10,4 µs |
| MicroLogix 1500 | Wort | 25,6 µs | 9,7 µs |
| | Doppelwort | 27,4 µs | 9,7 µs |

Mit dem LFU-Befehl werden bei einem Strompfadübergang von unwahr nach wahr Worte oder Doppelworte aus einem vom Anwender erstellten File, dem so genannten LIFO-Stapel, entladen. Die Daten werden entsprechend dem LIFO-Prinzip entladen. Im nachfolgend dargestellten LFL/LFU-Befehlspaar wurden Befehls- parameter programmiert.




**Befehlspaar LFL und LFU**




**Laden und Entladen des Stapels # N7:12**

Dieser Befehl verwendet folgende Operanden:

- LIFO - Der LIFO-Operand ist die Anfangsadresse des Stapels.
- Ziel - Der Zieloperand ist eine Wort- oder Doppelwortadresse, an der der Wert nach dem Entladen aus dem LIFO-Stapel gespeichert wird. Der LFU-Befehl entlädt diesen Wert aus dem letzten Standort des LIFO-Stapels und speichert ihn unter der Zieladresse. Die Adressierungsebene des Ziels muss mit der des LIFO-Stapels übereinstimmen. Wenn es sich bei LIFO um einen Wort-File handelt, muss das Ziel ein Wort-File sein. Wenn es sich bei LIFO um einen Doppelwort-File handelt, muss das Ziel ein Doppelwort-File sein.
- Steuerung - Dies ist die Adresse des Steuerfiles. Dieses Element enthält die Status-Bits, die Stapellänge und den Positionswert. Das Steuerelement besteht aus drei Worten:

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort 0** | -- | EU(1) | DN(2) | EM(3) | nicht belegt | | | | | | | | | | | |
| **Wort 1** | Länge - die maximale Anzahl Worte oder Doppelworte in dem Stapel | | | | | | | | | | | | | | | |
| **Wort 2** | Position - der nächste verfügbare Standort, an den Daten entladen werden können | | | | | | | | | | | | | | | |

(1) EU - Das FFU-Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde.

(2) DN - Fertig-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass der Stapel voll ist.

(3) EM - Leer-Bit; wenn dieses Bit gesetzt ist, bedeutet dies, dass der Stapel leer ist.

- Länge - Der Längenoperand enthält die Anzahl der Elemente in dem LIFO-Stapel. Die Länge des Stapels kann zwischen 1 und 128 (Wort) oder 1 und 64 (Doppelwort) betragen.
- Position - Dies ist der nächste Standort in dem LIFO-Stapel, aus dem Daten entladen werden. Die Position ist eine Komponente des Steuerregisters. Die Position kann zwischen 0 und 127 (Wort) oder 0 und 63 (Doppelwort) betragen. Die Position wird nach jedem Ladevorgang reduziert.

**Tabelle 14.19 Gültige Adressierungsmodi und Filetpyen für LFU-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

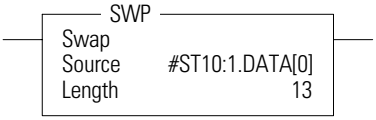
| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| LIFO | • | • | | • | | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Ziel | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Steuerung | | | | | (2) | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Position | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Nur Steuerfile. Gilt nicht für Zeitwerke und Zähler.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# SWP – Byte-Tausch




Befehlstyp: Ausgang

**Tabelle 14.20 Ausführungszeit des SWP-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 der Serie B und höher | 13,7 µs + 2,2 µs/getauschtes Wort | 0,0 µs |
| MicroLogix 1500 Serie B und höher | 11,7 µs + 1,8 µs/getauschtes Wort | 0,0 µs |

Verwenden Sie den SWP-Befehl zum Vertauschen der höher- und niederwertigen Bytes einer bestimmten Anzahl von Wörtern in einem Bit-, Ganzzahl- oder Zeichenkettenfile. Der SWP-Befehl verfügt über zwei Operanden:

- „Source“ (Quelle) entspricht der Wortadresse, die die zu vertauschenden Worte enthält.
- „Length“ (Länge) entspricht der Anzahl der zu vertauschenden Worte (unabhängig vom Filetyp). Die Adresse ist auf ganzzahlige Konstan- ten beschränkt. Für die Fileypten „Bit“ und „Ganzzahl“ beträgt die Länge zwischen 1 und 128. Für den Filetyp „Zeichenkette“ beträgt die Länge zwischen 1 und 41. Beachten Sie, dass dieser Befehl auf ein einzelnes Zeichenkettenelement beschränkt ist und nicht über die Grenze eines Zeichenkettenelements hinausgehen kann.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 14.21 Gültige Adressierungsmodi und Filetpyen für SWP-Befehl**

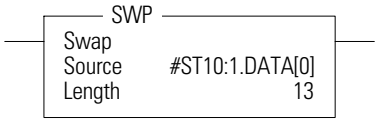
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden“ auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | | | | • | | • | | • | | | | | | | | | | | | | | | | | • | | | • | | |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig“) zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

**Beispiel:**




Quellenwert nach der Ausführung des SWP-Befehls:
abcdefghijklmnopqrstuvwxyzabcdefg
Quellenwert vor der Ausführung des SWP-Befehls:
badcfehgjilknmporqtsvuxwzyabcdefg

Die unterstrichenen Zeichen stellen die 13 Worte dar, bei denen das niederwertige Byte durch das höherwertige Byte vertauscht wurde.

# Schrittschaltwerksbefehle

Die Schrittschaltwerksbefehle steuern automatische Fertigungs- maschinen oder -prozesse, die sich ständig wiederholende Aufgaben ausführen. Sie sind in der Regel zeitgestützt oder ereignisgesteuert.

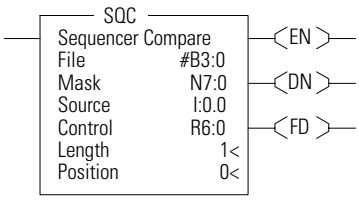
| Befehl | Zweck | Seite |
|---|---|---|
| SQC - Schrittschaltwerksvergleich | Vergleich von 16-Bit-Daten mit gespeicherten Daten | 15-2 |
| SQO - Schrittschaltwerksausgang | Übertragung von 16-Bit-Daten an Wortadressen | 15-6 |
| SQL - Schrittschaltwerksladung | Laden von 16-Bit-Daten in einen File | 15-9 |

Verwenden Sie die Schrittschaltwerksbefehle, um festzustellen, wann ein Schritt abgeschlossen ist. Mit dem SQO-Befehl (Schrittschaltwerks- ausgang) legen Sie für jeden Schritt die Ausgangsbedingungen fest. Verwenden Sie den SQL-Befehl (Schrittschaltwerksladung), um Daten in den Schrittschaltwerk-File zu laden.

Der wichtigste Vorteil der Schrittschaltwerksbefehle liegt in der effizienten Nutzung des Programmspeichers. Diese Befehle über- wachen und steuern gleichzeitig 16 (Wort) oder 32 (Doppelwort) diskrete Ausgänge in einem Strompfad.

Schrittschaltwerksbefehle können mit Bit-, Ganzzahl- oder Doppel-Ganzzahl-Files verwendet werden.

# SQC – Schrittschaltwerks-vergleich




Befehlstyp: Ausgang

**Tabelle 15.1 Ausführungszeit des SQC-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 23,5 µs | 7,1 µs |
| | Doppelwort | 26,3 µs | 7,1 µs |
| MicroLogix 1500 | Wort | 20,1 µs | 6,3 µs |
| | Doppelwort | 22,7 µs | 6,3 µs |

Verwenden Sie den SQC-Befehl zur Steuerung sequenzieller Maschinenfunktionen, um bei einem Strompfadübergang von unwahr nach wahr maskierte Quellenworte oder -doppelworte mit dem Maskenwert an der Referenzadresse (dem Schrittschaltwerks-File) zu vergleichen.

Wenn der Status aller nicht maskierter Bits in dem Quellenwort mit dem des entsprechenden Referenzworts übereinstimmt, wird in dem Steuerwort das Gefunden-Bit (FD) gesetzt. Wenn der Status nicht übereinstimmt, wird das Gefunden-Bit (FD) zurückgesetzt.

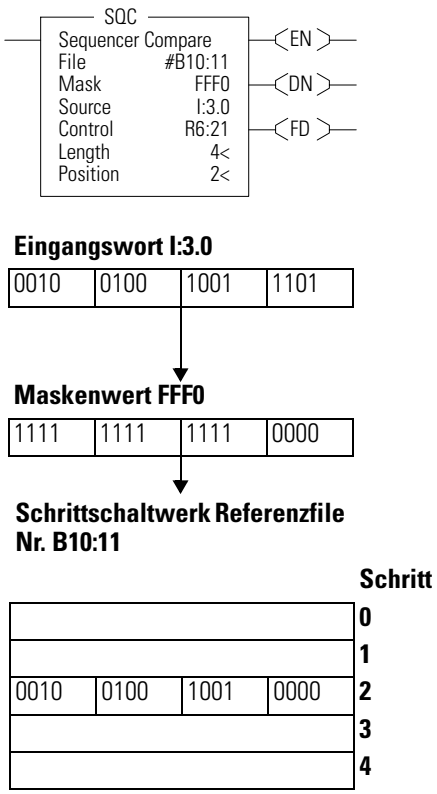
Sind die Bits zurückgesetzt (0), werden Daten ausmaskiert; sind Bits gesetzt (1), werden die Daten übertragen.

Die Maske kann fest oder variabel sein. Für eine feste Maske geben Sie einen Hexadezimalwert ein. Soll die Maske variabel sein und bei jedem Schritt geändert werden, kann eine Element- oder Fileadresse (direkt oder indirekt) eingegeben werden.

Bei einem Strompfadübergang von unwahr nach wahr wird der Befehl zum nächsten Schritt (Wort) in dem Schrittschaltwerks-File erhöht. Die in diesem File gespeicherten Daten werden durch eine Maske übertragen und mit der Quelle verglichen. Solange der Strompfad wahr ist, wird die Quelle bei jeder Abfrage mit den Referenzdaten verglichen. Wenn die Daten übereinstimmen, wird das FD-Bit in dem Steuerzähler des SQC-Befehls gesetzt.

Der SQC-Befehl kann unter anderem für Maschinendiagnosen eingesetzt werden.

Die folgende Abbildung zeigt, wie der SQC-Befehl ausgeführt wird.




Das FD-Bit des SQC-Befehls wird gesetzt, wenn eine Überein- stimmung zwischen einem Eingangswort und dem entsprechenden Referenzwort festgestellt wird.

In dem gezeigten Beispiel wird das FD-Bit R6:21/FD gesetzt, weil das Eingangswort dem Schrittschaltwerks-Referenzwert bei Verwendung des Maskenwerts entspricht.

Dieser Befehl verwendet folgende Operanden:

- File - Dies ist der Schrittschaltwerks-Referenzfile. Die Elemente dieses Files werden einzeln maskiert und mit dem in der Quelle gespeicherten Maskenwert verglichen.

  **HINWEIS** Wenn der Filetyp ein Wort ist, müssen auch die Maske und die Quelle Worte sein. Wenn der Filetyp ein Doppelwort ist, müssen auch die Maske und die Quelle Doppelworte sein.

- Maske - Der Maskenoperand enthält die Konstante, das Wort oder den File, die als Maske auf den File und die Quelle angewendet wird. Wenn ein Masken-Bit auf 1 gesetzt ist, werden die entsprechenden Daten für den Vergleich übertragen. Wenn ein Masken-Bit auf 0 gesetzt ist, werden die entsprechenden Daten ausmaskiert (nicht für den Vergleich übertragen). Die unmittel- baren Datenbereiche für die Maske sind 0 bis 0xFFFF oder 0 bis 0xFFFFFFFF.

  **HINWEIS** Bei direkter oder indirekter Maske wird der Standort des angegebenen Files über die Position ausgewählt.

- Quelle - Dies ist der Wert, der mit dem File verglichen wird.
- Steuerung - Dies ist die Adresse des Steuerfiles. Dieses Element enthält die Status-Bits, die Stapellänge und den Positionswert. Das Steuerelement besteht aus drei Worten:

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort 0** | EN(1) | -- | DN(2) | -- | ER(3) | nicht belegt | | FD(4) | nicht belegt | | | | | | | |
| **Wort 1** | Länge - enthält die Anzahl der Schritte in dem Schrittschaltwerks-Referenzfile | | | | | | | | | | | | | | | |
| **Wort 2** | Position - die aktuelle Position im Schrittschaltwerks-File | | | | | | | | | | | | | | | |

(1) EN - Das Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde.

(2) DN - Das Fertig-Bit wird gesetzt, wenn der Befehl für das letzte Wort in dem Schrittschaltwerks-File ausgeführt wurde. Dieses Bit wird beim nächsten Übergang des unwahren Strompfad nach wahr zurückgesetzt.

(3) ER - Das Fehler-Bit wird gesetzt, wenn ein negativer Positionswert festgestellt oder für die Länge ein negativer Wert oder null angegeben wurde. Wenn das ER-Bit gesetzt ist, wird auch das Bit für geringfügige Fehler (S2:5/2) gesetzt.

(4) FD - Das Gefunden-Bit wird gesetzt, wenn der Status aller nicht maskierter Bits in der Quellenadresse mit dem des Worts in dem Schrittschaltwerts-Referenzfile übereinstimmt. Ist der Strompfad wahr, wertet der Befehl dieses Bit bei jeder Ausführung aus.

- Länge - Der Längenoperand enthält die Anzahl der Schritte in dem Schrittschaltwerks-File (sowie der Maske und/oder Quelle, wenn es sich dabei um Filedaten handelt). Die Länge des Schrittschalt- werks-Files kann zwischen 1 und 256 betragen.
- Position - Dies ist der aktuelle Standort oder Schritt in dem Schrittschaltwerks-File (sowie der Maske und/oder Quelle, wenn es sich dabei um Filedaten handelt). Die Position bezeichnet den nächsten Standort im Stapel, an dem Daten des aktuellen Vergleichs gespeichert werden. Die Position ist eine Komponente des Steuerregisters. Gültige Werte für die Position sind 0 bis 255 bei Worten und 0 bis 127 bei Doppelworten. Die Position wird bei jedem Übergang von unwahr nach wahr erhöht.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 15.2 Gültige Adressierungsmodi und Filetpyen für SQC-Befehl**

*Definitionen der in dieser Tabelle verwendeten Befehle finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

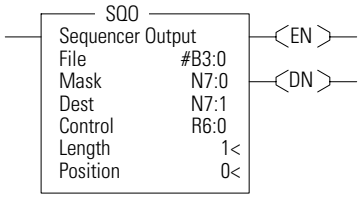
| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus[1] | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| File | • | • | | • | | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Maske | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Quelle | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Steuerung | | | | | (2) | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Position | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Nur Steuerfile.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressie- rung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# SQO – Schrittschaltwerksausgang




Befehlstyp: Ausgang

**Tabelle 15.3 Ausführungszeit des SQO-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 23,2 µs | 7,1 µs |
| | Doppelwort | 26,6 µs | 7,1 µs |
| MicroLogix 1500 | Wort | 20,0 µs | 6,3 µs |
| | Doppelwort | 23,1 µs | 6,3 µs |

Verwenden Sie den SQO-Befehl zur Steuerung sequenzieller Maschinenfunktionen, um bei einem Strompfadübergang von unwahr nach wahr maskierte Quellenreferenzworte oder -doppelworte an das Ziel zu übertragen. Bei einem Strompfadübergang von unwahr nach wahr wird der Befehl zum nächsten Schritt (Wort) in dem Schrittschalt- werks-File erhöht. Die für diesen Schritt gespeicherten Daten werden über eine Maske auf die vom Befehl spezifizierte Zieladresse übertragen. Bei jeder Ausführung des Befehls werden Daten in das Zielwort geschrieben.

Das Fertig-Bit wird nach der Übertragung des letzten Wortes eines Schrittschaltwerk-Files gesetzt. Beim nächsten Strompfadübergang von unwahr nach wahr wird die Position auf Schritt 1 zurückgesetzt.

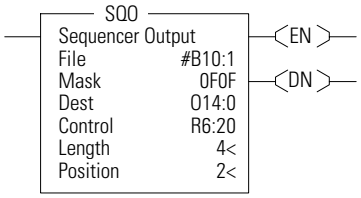
Lautet die Position beim Einschalten 0, hängt die weitere Funktions- weise des Befehls beim Umschalten von dem Programm-Modus in den Run-Modus davon ab, ob der Strompfad bei der ersten Abfrage wahr oder unwahr ist.

- Ist der Strompfad wahr, überträgt der Befehl den Wert von Schritt 0.
- Ist der Strompfad unwahr, wird der Wert von Schritt 1 erst beim ersten Strompfadübergang von unwahr nach wahr übertragen.

Sind die Bits zurückgesetzt (0), werden Daten ausmaskiert; sind Bits gesetzt (1), werden die Daten übertragen. Wenn Sie keine Masken-Bits setzen, ändert der Befehl keine Daten im Zielwort.

Die Maske kann fest oder variabel sein. Für eine feste Maske geben Sie einen Hexadezimalwert ein. Soll die Maske variabel sein und bei jedem Schritt geändert werden, kann eine Element- oder Fileadresse (direkt oder indirekt) eingegeben werden.

Die folgende Abbildung zeigt, wie der SQO-Befehl ausgeführt wird.




**Ziel O:14.0**

| 15 | 8 | 7 | 0 |
|---|---|---|---|
| 0000 | 0101 | 0000 | 1010 |

**Maskenwert 0F0F**

| 15 | 8 | 7 | 0 |
|---|---|---|---|
| 0000 | 1111 | 0000 | 1111 |

**Schrittschaltwerk Ausgangsfile Nr. B10:1**

| Wort | | | | | Schritt |
|---|---|---|---|---|---|
| B10:1 | 0000 | 0000 | 0000 | 0000 | 0 |
| B10:2 | 1010 | 0010 | 1111 | 0101 | 1 |
| B10:3 | 1111 | 0101 | 0100 | 1010 | 2 ← gegenwärtig ausgeführter Schritt |
| B10:4 | 0101 | 0101 | 0101 | 0101 | 3 |
| B10:5 | 0000 | 1111 | 0000 | 1111 | 4 |

**Externe Ausgänge (O:14) bei Schritt 2**

| | |
|---|---|
| 00 | |
| 01 | ON |
| 02 | |
| 03 | ON |
| 04 | |
| 05 | |
| 06 | |
| 07 | |
| 08 | ON |
| 09 | |
| 10 | ON |
| 11 | |
| 12 | |
| 13 | |
| 14 | |
| 15 | |

Dieser Befehl verwendet folgende Operanden:

- File - Dies ist der Schrittschaltwerks-Referenzfile. Die Elemente dieses Files werden einzeln maskiert und in der Zieladresse gespeichert.

  **HINWEIS** Wenn der Filetyp ein Wort ist, müssen auch die Maske und die Quelle Worte sein. Wenn der Filetyp ein Doppelwort ist, müssen auch die Maske und die Quelle Doppelworte sein.

- Maske - Der Maskenoperand enthält den Maskenwert. Wenn ein Masken-Bit auf 1 gesetzt ist, werden die entsprechenden Daten für den Vergleich übertragen. Wenn ein Masken-Bit auf 0 gesetzt ist, werden die entsprechenden Daten ausmaskiert (nicht ins Ziel übertragen). Die unmittelbaren Datenbereiche für die Maske sind 0 bis 0xFFFF (Wort) oder 0 bis 0xFFFFFFFF (Doppelwort).

  **HINWEIS** Bei direkter oder indirekter Maske wird der Standort des angegebenen Files über die Position ausgewählt.

- Ziel - Der Zieloperand ist der Schrittschaltwerks-File.
- Steuerung - Dies ist die Adresse des Steuerfiles. Dieses Element enthält die Status-Bits, die Stapellänge und den Positionswert. Das Steuerelement besteht aus drei Worten:

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort 0** | EN(1) | -- | DN(2) | -- | ER(3) | nicht belegt | | FD | nicht belegt | | | | | | | |
| **Wort 1** | Länge - enthält den Index des letzten Elements in dem Schrittschaltwerks-Referenzfile. | | | | | | | | | | | | | | | |
| **Wort 2** | Position - die aktuelle Position im Schrittschaltwerks-File | | | | | | | | | | | | | | | |

(1) EN - Das Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde.

(2) DN - Das Fertig-Bit wird gesetzt, wenn der Befehl für das letzte Wort in dem Schrittschaltwerks-File ausgeführt wurde. Dieses Bit wird beim nächsten Übergang des unwahren Strompfad nach wahr zurückgesetzt.

(3) ER - Das Fehler-Bit wird gesetzt, wenn ein negativer Positionswert festgestellt oder für die Länge ein negativer Wert oder null angegeben wurde. Wenn das ER-Bit gesetzt ist, wird auch das Bit für geringfügige Fehler (S2:5/2) gesetzt.

- Länge - Der Längenoperand enthält die Anzahl der Schritte in dem Schrittschaltwerks-File (sowie der Maske und/oder dem Ziel, wenn es sich dabei um Filedaten handelt). Die Länge des Schrittschaltwerks-Files kann zwischen 1 und 256 betragen.
- Position - Dies ist der aktuelle Standort oder Schritt in dem Schrittschaltwerks-File (sowie der Maske und/oder des Ziels, wenn es sich dabei um Filedaten handelt). Die Position bezeichnet den nächsten Standort im Stapel, der maskiert und an das Ziel übertragen wird. Die Position ist eine Komponente des Steuer- registers. Gültige Positionswerte liegen zwischen 0 und 255. Die Position wird bei jedem Übergang von unwahr nach wahr erhöht.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 15.4 Gültige Adressierungsmodi und Filetypen für SQO-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Element | Bit | Wort | Doppelwort |
| File(2) | • | • | | • | | • | | | • | | | | | | | | | | | | | | | | • | • | | | • | • |
| Maske(2) | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | • | • | • | | | • | • |
| Zieladresse(2) | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | | • | • | | | • | • |
| Steuerung | | | | | (3) | | | | | | | | | | | | | | | | | | | | • | | • | | | |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | | • | |
| Position | | | | | | | | | | | | | | | | | | | | | | | | • | | | | | • | |

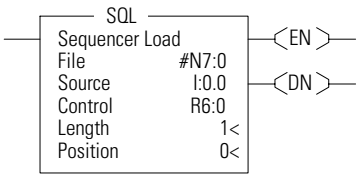
(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

(2) Auch die direkte und indirekte Fileadressierung können verwendet werden.

(3) Nur Steuerfile.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressie- rung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# SQL – Schrittschaltwerksladung




Befehlstyp: Ausgang

"

**Tabelle 15.5 Ausführungszeit des SQL-Befehls**

| Steuerung | Datengröße | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 21,7 µs | 7,0 µs |
| | Doppelwort | 24,3 µs | 7,1 µs |
| MicroLogix 1500 | Wort | 19,1 µs | 6,3 µs |
| | Doppelwort | 21,1 µs | 6,3 µs |

Verwenden Sie den SQL-Befehl, um bei einem Strompfadübergang von unwahr nach wahr bei jedem Schritt einer Schrittschaltwerks- operation Worte oder Doppelworte in einen Schrittschaltwerks-File zu laden. Dieser Befehl verwendet folgende Operanden:

- File - Dies ist der Schrittschaltwerks-Referenzfile. Die Elemente dieses Files werden einzeln von der Quelle übertragen.

  **HINWEIS** Wenn der Filetyp ein Wort ist, müssen auch die Maske und die Quelle Worte sein. Wenn der Filetyp ein Doppelwort ist, müssen auch die Maske und die Quelle Doppelworte sein.

- Quelle - Der Quellenoperand ist eine Konstante oder eine Adresse mit dem Wert, der an die momentan verfügbare Position in dem Schrittschaltwerks-File gestellt werden soll. Die Adressierungs- ebene der Quelle muss mit der des Schrittschaltwerks-Files übereinstimmen. Wenn der Filetyp ein Wort ist, muss auch die Quelle ein Wort sein. Wenn der Filetyp ein Doppelwort ist, muss auch die Quelle ein Doppelwort sein. Die Quellenwerte müssen in dem Bereich -32768 bis 32767 (Wort) oder -2.147.483.648 bis 2.147.483.647 (Doppelwort) liegen.
- Steuerung - Dies ist die Adresse des Steuerfiles. Dieses Element enthält die Status-Bits, die Stapellänge und den Positionswert. Das Steuerelement besteht aus drei Worten:

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort 0** | EN(1) | -- | DN(2) | -- | ER(3) | nicht belegt | | FD | nicht belegt | | | | | | | |
| **Wort 1** | Länge - enthält den Index des letzten Elements in dem Schrittschaltwerks-Referenzfile. | | | | | | | | | | | | | | | |
| **Wort 2** | Position - die aktuelle Position im Schrittschaltwerks-File | | | | | | | | | | | | | | | |

(1) EN - Das Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde.

(2) DN - Das Fertig-Bit wird gesetzt, wenn der Befehl für das letzte Wort in dem Schrittschaltwerks-File ausgeführt wurde. Dieses Bit wird beim nächsten Übergang des unwahren Strompfad nach wahr zurückgesetzt.

(3) ER - Das Fehler-Bit wird gesetzt, wenn ein negativer Positionswert festgestellt oder für die Länge ein negativer Wert oder null angegeben wurde. Wenn das ER-Bit gesetzt ist, wird auch das Bit für geringfügige Fehler (S2:5/2) gesetzt.

- Länge - Der Längenoperand enthält die Anzahl der Schritte in dem Schrittschaltwerks-File (dies ist auch die Länge der Quelle, wenn es sich dabei um Filedaten handelt). Die Länge des Schrittschalt- werks-Files kann zwischen 1 und 256 betragen.
- Position - Dies ist der aktuelle Standort oder Schritt in dem Schrittschaltwerks-File (sowie der Quelle, wenn es sich dabei um Filedaten handelt). Die Position bezeichnet den nächsten verfügbaren Standort im Stapel, an dem der Wert oder die Konstante aus der Quelle gespeichert wird. Die Position ist eine Komponente des Steuerregisters. Gültige Positionswerte liegen zwischen 0 und 255.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 15.6 Gültige Adressierungsmodi und Filetypen für SQL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus(1) | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| File(2) | • | • | | • | | • | | | • | | | | | | | | | | | | | | | | • | • | | • | • | |
| Quelle(2) | • | • | | • | | • | | | • | | | | | | | | | | | | | | | • | • | • | | • | • | |
| Steuerung | | | | | (3) | | | | | | | | | | | | | | | | | | | | • | | | | | • |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Position | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

(1) Bitte beachten Sie den nachfolgenden Hinweis ("Wichtig") zur indirekten Adressierung.

(2) Auch die direkte und indirekte Fileadressierung kann verwendet werden.

(3) Nur Steuerfile.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressie- rung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# Programmsteuerungsbefehle

Mit diesen Befehlen können Sie die Reihenfolge der Abfrage eines Kontaktplans durch den Prozessor ändern. In der Regel werden diese Befehle verwendet, um die Abfragezeit zu minimieren, die Effizienz des Programms zu steigern und Fehler in einem Kontaktplan zu beheben.

| Befehl | Zweck | Seite |
|---|---|---|
| JMP - Sprung zu Marke | Sprung nach vorn oder hinten zum entsprechenden Sprungmarkenbefehl | 16-1 |
| LBL - Marke | | 16-2 |
| JSR - Sprung ins Unterprogramm | Sprung zum angegebenen Unterprogramm und Rückkehr zum Hauptprogramm | 16-2 |
| SBR - Unterprogramm | | 16-3 |
| RET - Rückkehr vom Unterprogramm | | 16-3 |
| SUS - Suspend | Fehler im Anwenderprogramm beheben oder diagnostizieren | 16-4 |
| TND - Temporäres Ende | Aktuelle Kontaktplanabfrage beenden | 16-4 |
| END - Programmende | Ein Programm oder ein Unterprogramm beenden | 16-5 |
| MCR - Hauptsteuerbefehl | Hauptsteuerzone im Kontaktplan aktivieren oder deaktivieren | 16-5 |

## JMP – Sprung zu Marke

Befehlstyp: Ausgang

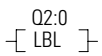
**Tabelle 16.1 Ausführungszeit des JMP-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 1,0 µs | 0,0 µs |
| MicroLogix 1500 | 1,0 µs | 0,0 µs |

Mit dem JMP-Befehl wird die Ausführreihenfolge des Kontaktplans durch die Steuerung geändert. Bei einem Sprung wird die Ausführung des Programms an dem gekennzeichneten Strompfad (LBL *Markennummer*) fortgesetzt. Sprünge können in der Kontaktplanlogik eines Programmfiles vorwärts oder rückwärts erfolgen. Mehrere JMP-Befehle können auf dieselbe Marke zur Fortsetzung der Ausführung verweisen.

Gültige unmittelbare Wert für die Marke liegen zwischen 0 und 999. Die Marke ist Teil des Programm-Files.

# LBL – Marke




Befehlstyp: Eingang

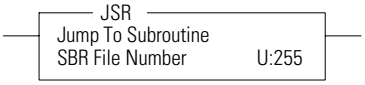
**Tabelle 16.2 Ausführungszeit des LBL-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 1,0 µs | 1,0 µs |
| MicroLogix 1500 | 1,0 µs | 1,0 µs |

Der LBL-Befehl wird zusammen mit dem JMP-Befehl (Sprung zu Marke) verwendet, um die Ausführreihenfolge eines Kontaktplans zu ändern. Bei einem Sprung wird die Ausführung des Programms an dem gekennzeichneten Strompfad (LBL *Markennummer*) fortgesetzt.

Gültige unmittelbare Wert für die Marke liegen zwischen 0 und 999. Die Marke ist Teil des Programm-Files.

# JSR – Sprung ins Unterprogramm




Befehlstyp: Ausgang

**Tabelle 16.3 Ausführungszeit des JSR-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 8,4 µs | 0,0 µs |
| MicroLogix 1500 | 8,0 µs | 0,0 µs |

Mit dem JSR-Befehl wird die Ausführung eines separaten Unterprogramm-Files innerhalb eines Kontaktplans gestartet. Dabei wird die Ausführung des Programms mit dem angegebenen Unterpro- gramms (SBR *Filenummer*) fortgesetzt. Nach Ausführung des Unter- programms wird die Ausführung mit dem Befehl fortgesetzt, der auf den JSR-Befehl folgt.

Gültige unmittelbare Werte für den JSR-File liegen zwischen 3 und 255.

# SBR – Unterprogramm

Befehlstyp: Eingang

**Tabelle 16.4 Ausführungszeit des SBR-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 1,0 µs | 1,0 µs |
| MicroLogix 1500 | 1,0 µs | 1,0 µs |

Der SBR-Befehl ist eine Marke, die nicht von dem Prozessor verwendet wird. Diese Marke dient nur der Kennzeichnung des ersten Strompfads eines Unterprogramms. Dieser Befehl ist der erste Befehl auf einem Strompfad und wird immer als wahr bewertet.

# RET – Rückkehr vom Unterprogramm

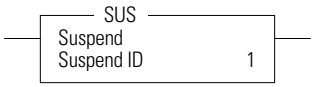
Befehlstyp: Ausgang

**Tabelle 16.5 Ausführungszeit des RET-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 1,0 µs | 0,0 µs |
| MicroLogix 1500 | 1,0 µs | 0,0 µs |

Der RET-Befehl kennzeichnet das Ende der Ausführung eines Unterprogramms oder das Ende des Unterprogramm-Files. Bei diesem Befehl wird die Ausführung des Kontaktplans mit dem Befehl fortgesetzt, der dem JSR-Befehl, dem Anwender-Interrupt oder der Anwender-Fehlerroutine folgt, mit dem/der die Ausführung dieses Unterprogramms ausgelöst wurde.

# SUS – Suspend




Befehlstyp: Ausgang

Der SUS-Befehl kann zur Erfassung und Identifizierung bestimmter Zustände beim Austesten des Programms für die Fehlersuche und -beseitigung innerhalb des Gesamtsystems verwendet werden. Dabei wird der Prozessor in den Suspend- oder Leerlaufmodus gesetzt, bei dem die Spannungsversorgung zu allen Ausgängen unterbrochen ist. Die Unterbrechungskennung und der Unterbrechungsfile (Programm- oder Unterprogrammnummer, die angibt, wo der SUS-Befehl gespeichert ist) werden in den Statusfile (S:7 und S:8) gestellt.

Gültige unmittelbare Werte für die Unterbrechungskennung liegen zwischen -32768 und 32767.

# TND – Temporäres Ende




Befehlstyp: Ausgang

**Tabelle 16.6 Ausführungszeit des TND-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 0,9 µs | 0,0 µs |
| MicroLogix 1500 | 1,0 µs | 0,0 µs |

Mit dem TND-Befehl wird das vorzeitige Ende der Ausführung eines Kontaktplans herbeigeführt. In einem STI-, HSC- oder EII-Unter- programm oder einem Anwenderfehler-Unterprogramm kann der TND-Befehl nicht ausgeführt werden. Dieser Befehl kann in einem Kontaktplan mehrfach verwendet werden.

In einem wahren Strompfad verhindert der TND-Befehl die Abfrage des restlichen Programmfiles durch den Prozessor. Darüber hinaus löst dieser Befehl die Ausgangsabfrage, die Eingangsabfrage und die Verwaltung des Prozessorscanzyklus vor der Wiederaufnahme der Abfrage an Strompfad 0 des Hauptprogramms (File 2) aus. Bei Ausführung dieses Befehls in einem verschachtelten Unterprogramm wird die Ausführung aller verschachtelten Unterprogramme beendet.

# END – Programmende

–( END )–

Befehlstyp: Ausgang

Der END-Befehl muss am Ende jedes Kontaktplanprogramms stehen. Im Hauptprogrammfile (File 2) beendet dieser Befehl die Programm- abfrage. In einem Unterprogramm-, Interrupt- Anwenderfehler-File veranlasst der END-Befehl eine Rückkehr vom Unterprogramm.

# MCR – Hauptsteuerbefehl

–( MCR )–

Befehlstyp: Ausgang

**Tabelle 16.7 Ausführungszeit für MCR-Befehl**

| Steuerung | Befehl | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | MCR-Start | 1,2 µs | 1,2 µs |
| | MCR-Ende | 1,6 µs | 1,6 µs |
| MicroLogix 1500 | MCR-Start | 0,8 µs | 0,8 µs |
| | MCR-Ende | 1,0 µs | 1,0 µs |

Der MCR-Befehl wird paarweise zur Steuerung der Kontaktplanlogik zwischen den beiden Einzelbefehlen des Paars verwendet. Strompfade innerhalb der MCR-Zone werden weiterhin abgefragt, jedoch wird die Abfragezeit durch den unwahren Zustand der nicht speichernden Ausgänge reduziert. Nicht speichernde Ausgänge werden zurückge- setzt, sobald der Strompfad unwahr wird.

Mit diesem Befehl werden die Grenzen einer MCR-Zone festgelegt. Eine MCR-Zone umfasst die Befehle der Kontaktplanlogik, die zwischen den beiden Einzelbefehlen eines MCR-Befehlspaars liegen. Der Anfang einer MCR-Zone ist als der Strompfad definiert, der den MCR-Befehl enthält, der auf eine bedingte Logik folgt. Das Ende einer MCR-Zone ist als der erste Strompfad definiert, der nur einen MCR-Befehl enthält, der nach einem Strompfad zur Kennzeichnung des Anfangs einer MCR-Zone folgt (siehe unten).

Wenn der Strompfadstatus des ersten MCR-Befehls wahr ist, wird die Zone bei der Ausführung ignoriert. Wenn der Strompfadstatus des ersten MCR-Befehls unwahr ist, wird die Kontaktplanlogik innerhalb der MCR-Zone wie bei einem unwahren Strompfad ausgeführt. Alle nicht speichernden Ausgänge innerhalb der MCR-Zone werden zurückgesetzt.

MCR-Bereiche ermöglichen die Aktivierung bzw. Sperrung bestimmter Programmabschnitte, z. B. bei Rezeptanwendungen.

Beachten Sie bei der Programmierung von MCR-Befehlen Folgendes:

- Sie müssen den Bereich mit einem nicht bedingten MCR-Befehl abschließen.
- Sie können MCR-Bereiche nicht verschachteln.
- Sie sollten keinen Sprung in einen MCR-Bereich programmieren. Ist der Bereich unwahr, so wird er dadurch aktiviert.

**HINWEIS** Der MCR-Befehl ist kein Ersatz für ein festverdrah- tetes Hauptsteuerrelais, das Notabschaltungsfunk- tionen erfüllt. Sie müssen in jedem Fall ein festverdrahtetes Hauptsteuerrelais für Notabschaltungen der E/A-Spannungsversorgung installieren.

**ACHTUNG**

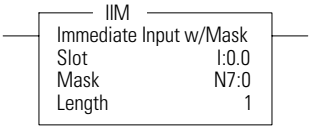
Wenn Sie innerhalb eines MCR-Bereichs Befehle wie Zeitwerke oder Zähler starten, wird die Befehls- ausführung beendet, nachdem der Bereich deaktiviert wird. Programmieren Sie daher kritische Operationen außerhalb des Bereichs ggf. neu.

# Ein- und Ausgangsbefehle

Mit den Eingangs- und Ausgangsbefehlen können Sie Daten gezielt und ohne Unterstützung durch die Eingangs- und Ausgangsabfragen aktualisieren.

| Befehl | Zweck | Seite |
|---|---|---|
| IIM - Sofortiger Eingang mit Maske | Daten vor der normalen Eingangsabfrage aktualisieren. | 17-1 |
| IOM - Sofortiger Ausgang mit Maske | Ausgänge vor der normalen Ausgangsabfrage aktualisieren. | 17-4 |
| REF - E/A-Auffrischung | Programmabfrage für die Ausführung der E/A-Abfrage (Ausgänge schreiben, Kommunikation bearbeiten, Eingänge lesen) unterbrechen. | 17-5 |

## IIM – Sofortiger Eingang mit Maske




Befehlstyp: Ausgang

**HINWEIS** Dieser Befehl ist nur für integrierte E/A verfügbar. Für Erweiterungs-E/A kann dieser Befehl nicht eingesetzt werden.

**Tabelle 17.1 Ausführungszeit des IIM-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 26,4 µs | 0,0 µs |
| MicroLogix 1500 | 22,5 µs | 0,0 µs |

Mit dem IIM-Befehl können Sie Eingangsdaten gezielt und ohne Unterstützung durch die automatische Eingangsabfrage aktualisieren. Dieser Befehl verwendet folgende Operanden:

- **Steckplatz** – Dieser Operand bezeichnet den Standort der Daten, die zur Aktualisierung des Eingangsfiles verwendet werden können. Dabei werden die Steckplatz- und die Wortnummer der betreffenden Daten angegeben. Beispiel: Bei Slot = I:0 werden Eingangsdaten von Steckplatz 0 ab Wort 0 maskiert und in den Eingangsdatenfile I:0, beginnend bei Wort 0 und mit der angegebenen Länge, gestellt. Bei Slot = I0.1 wird Wort 1 von Steckplatz 0 verwendet, usw.

**WICHTIG** Slot 0 ist bei diesem Befehl die einzige gültige Steckplatznummer. Der IIM-Befehl kann nicht für Erweiterungs-E/A verwendet werden.

- **Maske** – Die Maske ist eine hexadezimale Konstante oder Registeradresse, die den Maskenwert enthält, der auf den Steckplatz angewendet werden soll. Ist ein Masken-Bit auf 1 gesetzt, werden die entsprechenden Bitdaten von dem Steckplatz an den Eingangsdatenfile übertragen. Ist ein Masken-Bit auf 0 gesetzt, wird die Übertragung der entsprechenden Bitdaten von dem Steckplatz an den Eingangsdatenfile gesperrt. Der Masken- wert kann zwischen 0 und 0xFFFF liegen.

| **Bit** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Physischer Eingang | Eingangswort | | | | | | | | | | | | | | | |
| Maske | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Eingangs-datenfile | Daten werden nicht aktualisiert | | | | | | | | Daten werden entsprechend Eingangswort aktualisiert | | | | | | | |

- **Länge** – Dies ist die Anzahl der maskierten Worte, die an den Eingangsdatenfile übertragen werden.

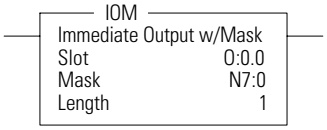
Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 17.2 Gültige Adressierungsmodi und Filetypen für IIM-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Steckplatz | | • | | | | | | | | | | | | | | | | | | | | | | | • | | | • | | |
| Maske | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | • | • | | • | | |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | | | |

# IOM – Sofortiger Ausgang mit Maske




Befehlstyp: Ausgang

**HINWEIS** Dieser Befehl ist nur für integrierte E/A verfügbar. Für Erweiterungs-E/A kann dieser Befehl nicht eingesetzt werden.

**Tabelle 17.3 Ausführungszeit des IOM-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 22,3 µs | 0,0 µs |
| MicroLogix 1500 1764-LSP | 18,4 µs | 0,0 µs |
| MicroLogix 1500 1764-LRP | 19,4 µs | 0,0 µs |

Mit dem IOM-Befehl können Sie Ausgangsdaten gezielt und ohne Unterstützung durch die automatische Ausgangsabfrage aktualisieren. Dieser Befehl verwendet folgende Operanden:

- **Steckplatz** – Der Steckplatz bezeichnet den physischen Standort der Daten, die anhand von Daten aus dem Ausgangsfile aktualisiert werden.

**WICHTIG** Slot 0 ist bei diesem Befehl die einzige gültige Steckplatznummer. Der IOM-Befehl kann nicht für Erweiterungs-E/A verwendet werden.

- **Maske** – Die Maske ist eine hexadezimale Konstante oder Registeradresse, die den Maskenwert enthält, der angewendet werden soll. Ist ein Masken-Bit auf 1 gesetzt, werden die entsprechenden Bitdaten an die physischen Ausgänge übertragen. Ist ein Masken-Bit auf 0 gesetzt, wird die Übertragung der entsprechenden Bitdaten an die Ausgänge gesperrt. Der Maskenwert kann zwischen 0 und 0xFFFF liegen.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Ausgangs-daten | Ausgangswort | | | | | | | | | | | | | | | |
| Maske | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Physische Ausgänge | Daten werden nicht aktualisiert | | | | | | | | Daten werden entsprechend Ausgangswort aktualisiert | | | | | | | |

- **Länge** – Dies ist die Anzahl der maskierten Worte, die an die Ausgänge übertragen werden.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 17.4 Gültige Adressierungsmodi und Filetypen für IOM-Befehl**
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Steckplatz | • | | | | | | | | | | | | | | | | | | | | | | | | • | | | • | | |
| Maske | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | • | • | | • | | |
| Länge | | | | | | | | | | | | | | | | | | | | | | | | • | | | | | | |

# REF – E/A auffrischen

Befehlstyp: Ausgang

**Tabelle 17.5 Ausführungszeit des REF-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | siehe S. A-7 | 0,0 µs |
| MicroLogix 1500 | siehe S. B-7 | 0,0 µs |

Der REF-Befehl wird zum Unterbrechen der Programmabfrage verwendet, um die E/A-Scannung und Kommunikationsbearbeitung innerhalb des aktuellen Zyklus für alle Kommunikationskanäle zu unterbrechen. Hierzu gehören u. a.: Ausgänge schreiben, Kommuni- kation bearbeiten (alle Kommunikationskanäle, Kommunikations- schalter, DAT *[nur MicroLogix 1500]* und Kommunikationsverwaltung) und Eingänge lesen.

Der REF-Befehl hat keine Programmierparameter. Wenn dieser Befehl als wahr erkannt wird, wird die Programmabfrage für die E/A-Abfrage und Kommunikationsbearbeitung innerhalb des aktuellen Zyklus unterbrochen. Die Abfrage wird dann mit dem Befehl, der dem REF-Befehl folgt, fortgesetzt.

In einem STI-, HSC- oder EII-Unterprogramm oder einem Anwenderfehler-Unterprogramm kann der REF-Befehl nicht ausgeführt werden.

**HINWEIS** Ein REF-Befehl kann zur Änderung von Eingangs- daten während einer Programmabfrage führen. Dieser Umstand muss bei der Ausführung eines REF-Befehls beachtet werden.

**ACHTUNG**

Die Watchdog- und Abfragezeitwerke werden bei der Ausführung des REF-Befehls zurückgesetzt. Stellen Sie sicher, dass sich der REF-Befehl nicht in einer Endlos-Programmschleife befindet. Fügen Sie den REF-Befehl nur in eine Programmschleife ein, wenn Sie das Programm analysiert haben.

# Interrupts verwenden

Mit Hilfe von Interrupts können Sie Programme auf der Grundlage definierter Ereignisse unterbrechen. Dieses Kapitel enthält Informationen über die Verwendung von Interrupts, Interrupt- Befehlen und Interrupt-Funktionsfiles. Diese Themen werden wie folgt behandelt:


- „Informationen zur Verwendung von Interrupts“ auf Seite 18-2.
- „Anwender-Interrupt-Befehle“ auf Seite 18-7.
- „STI-Funktionsfile verwenden“ auf Seite 18-12.
- „EII-Funktionsfile verwenden“ auf Seite 18-17.


Siehe auch: „Verwenden des Hochgeschwindigkeitszählers und des programmierbaren Endschalters“ auf Seite 5-1.

# Informationen zur Verwendung von Interrupts

Die grundlegenden Merkmale der Anwender-Interrupts werden in diesem Abschnitt unter den folgenden Überschriften erläutert:

- Was ist ein Interrupt?
- Wann kann der Betrieb der Steuerung unterbrochen werden?
- Priorität bei Anwender-Interrupts
- Interrupt-Latenzzeit
- Anwenderfehlerroutine

## Was ist ein Interrupt?

Ein Interrupt ist ein Ereignis, das zur Unterbrechung der momentan durch die Steuerung ausgeführten Task, zur Durchführung einer anderen Task und zur anschließenden Fortsetzung der unterbroche- nen Task an der Stelle der Unterbrechung führt. MicroLogix 1200- und MicroLogix 1500-Steuerungen unterstützen folgende Anwender-Interrupts:

- Fehlerbehandlungsroutine
- Ereignis-Interrupts (4)
- Hochgeschwindigkeitszähler-Interrupts(1)
- Wählbare zeitgesteuerte Interrupts

Interrupts müssen konfiguriert und zur Ausführung aktiviert werden. Bei Konfiguration, Aktivierung und anschließender Ausführung eines Interrupts geschieht Folgendes:

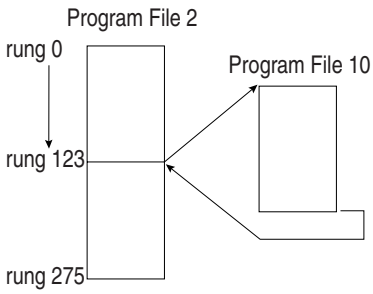
1. Die Ausführung des Anwenderprogramms wird unterbrochen.
2. Das Anwenderprogramm führt eine definierte Task durch, die die Grundlage des Interrupts bildet.
3. Das Anwenderprogramm kehrt zu der unterbrochenen Aufgabe zurück.

**Funktionsweise eines Interrupts, Beispiel**

Programmfile 2 ist das Hauptsteuerprogramm.

Programmfile 10 ist die Interruptroutine.

- In Strompfad 123 tritt ein Interrupt-Ereignis auf.
- Programmfile 10 wird ausgeführt.
- Die Ausführung von Programmfile 2 wird direkt nach Abfrage von Programmfile 10 wieder aufgenommen.




(1) Die MicroLogix 1200-Steuerung verfügt über einen HSC-Interrupt, HSC0. Die MicroLogix 1500-Steuerung verfügt über zwei HSC-Interrupts, HSC0 und HSC1.

Bei normaler Programmausführung durch die Steuerung und Eintritt eines Interrupt-Ereignisses geschieht Folgendes:

1. Die normale Ausführung durch die Steuerung wird gestoppt.
2. Der Interrupt wird durch die Steuerung identifiziert.
3. Die Steuerung springt unverzüglich zu Strompfad 0 des Unterprogramms, das für den Anwender-Interrupt angegeben ist.
4. Die Ausführung des Anwender-Interrupt-Unterprogramms (oder einer Reihe von Unterprogrammen, falls das angegebene Unterprogramm auf weitere Unterprogramme verweist) wird gestartet.
5. Das/Die Unterprogramm(e) wird (werden) vollständig abgeschlossen.
6. Der Normalbetrieb wird an dem Punkt wieder aufgenommen, an dem das Steuerprogramm unterbrochen worden war.

## Wann kann der Betrieb der Steuerung unterbrochen werden?

Bei MicroLogix 1200- und MicroLogix 1500-Steuerungen können Interrupts nur während bestimmter Phasen einer Programmabfrage bearbeitet werden. Hierzu gehören:

- der Beginn eines Kontaktplanstrompfads
- jederzeit während Abfrageende
- zwischen Datenworten bei einer Erweiterungs-E/A-Abfrage

Nur in diesen Phasen wird ein Interrupt durch die Steuerung bear- beitet. Ist der Interrupt deaktiviert, wird beim nächsten Eintreten einer der drei oben genannten Möglichkeiten das Anstehend-Bit gesetzt.

**ACHTUNG**

Wenn Interrupts während der Programmabfrage über einen OTL-, OTE- oder UIE-Befehl aktiviert werden, *muss* dieser Befehl der *letzte* Befehl sein, der auf dem Strompfad ausgeführt wird (letzter Befehl auf letztem Abzweig). Es wird empfohlen, in dem Strompfad diesen Befehl als einzigen Ausgangsbefehl zu verwenden.

## Priorität bei Anwender-Interrupts

Mehrere Interrupts werden entsprechend ihrer jeweiligen Priorität bearbeitet.

Wenn ein Interrupt eintritt und andere Interrupts bereits eingetreten sind, aber noch nicht bearbeitet wurden, wird der neue Interrupt entsprechend seiner Priorität gegenüber den anderen anstehenden Interrupts zur Ausführung eingeplant. Bei der nächsten Möglichkeit zur Bearbeitung von Interrupts werden alle Interrupts in der Reihenfolge ihrer Priorität (von höchster zu niedrigster Priorität) bearbeitet.

Tritt ein Interrupt während der Bearbeitung (Ausführung) eines Interrupts mit niedriger Priorität ein, wird die aktuell ausgeführte Interrupt-Routine unterbrochen und der Interrupt mit höherer Priorität bearbeitet. Danach wird der Interrupt mit niedriger Priorität abgeschlossen und der Normalbetrieb wieder aufgenommen.

Tritt ein Interrupt während der Bearbeitung (Ausführung) eines Interrupts mit höherer Priorität ein und wurde das Anstehend-Bit für den Interrupt mit niedriger Priorität gesetzt, wird die aktuell ausge- führte Interrupt-Routine vollständig ausgeführt. Danach wird der Interrupt mit niedriger Priorität ausgeführt und der Normalbetrieb wieder aufgenommen.

Reihenfolge der Prioritäten:

| | |
|---|---|
| Anwenderfehlerroutine | **höchste Priorität** |
| Ereignis-Interrupt 0 | |
| Ereignis-Interrupt 1 | |
| Hochgeschwindigkeitszähler-Interrupt 0 | |
| Ereignis-Interrupt 2 | |
| Ereignis-Interrupt 3 | |
| Hochgeschwindigkeitszähler-Interrupt 1 (*nur MicroLogix 1500*) | |
| Wählbarer, zeitgesteuerter Interrupt | **niedrigste Priorität** |

## Interrupt-Latenzzeit

Die Interrupt-Latenzzeit ist definiert als die maximale Zeitspanne, die zwischen dem Eintritt eines Interrupts und dem Beginn der Ausfüh- rung des Interrupt-Unterprogramms vergeht. Die folgenden Tabellen zeigen den Zusammenhang zwischen einem Interrupt und dem Betriebszyklus der Steuerung.

| Programmabfrageaktivität | Möglicher Eintritt eines Interrupts |
|---|---|
| Eingangsabfrage | zwischen Wortaktualisierungen |
| Kontaktplanabfrage | Start des Strompfads |
| Ausgangsabfrage | zwischen Wortaktualisierungen |
| Kommunikationsbearbeitung | jederzeit(1)(2) |
| Verwaltung | jederzeit |

(1) Der Kommunikationsdienst beinhaltet 80 µs für den Übergang in ein Unterprogramm.

(2) Der Kommunikationsdienst beinhaltet 60 µs für einen Zeittakt.

Die Interrupt-Latenzzeit können Sie wie folgt ermitteln:

1. Bestimmen Sie zunächst die Ausführungszeit des Strompfads mit der längsten Ausführungszeit im Steuerprogramm (maximale Strompfadzeit). Weitere Informationen hierzu finden Sie unter „MicroLogix 1500 – Speicherbelegung und Befehlsausführungszeit“ auf Seite B-1 oder „MicroLogix 1500 – Speicherbelegung und Befehlsausführungszeit“ auf Seite B-1.
2. Multiplizieren Sie die maximale Strompfadzeit mit dem Kommu- nikations-Multiplikator, der Ihrer Konfiguration im Abschnitt „MicroLogix 1200 – Arbeitsblatt zur Abfragezeit“ auf Seite A-7 oder „MicroLogix 1500 – Arbeitsblatt zur Abfragezeit“ auf Seite B-7 entspricht.

Das Ergebnis dieser Berechnung ist wie folgt zu verstehen:

| Steuerung | Wenn die in Schritt 2 berechnete Zeit: | Dann ist die Interrupt-Latenzzeit: |
|---|---|---|
| MicroLogix 1200 | kleiner als 133 µs ist | 411 µs |
| | größer als 133 µs ist | der in Schritt 2 berechnete Wert plus 278 µs |
| MicroLogix 1500 | kleiner als 100 µs ist | 360 µs |
| | größer als 100 µs ist | der in Schritt 2 berechnete Wert plus 260 µs |

## Anwenderfehlerroutine

Die Fehlerbehandlungsroutine bietet dem Anwender die Möglichkeit, eine Außerbetriebsetzung der Steuerung zu verhindern, falls bestimm- te Anwenderfehler auftreten. Die Fehlerbehandlungsroutine wird ausgeführt, wenn ein behebbarer oder nicht behebbarer Fehler auftritt. Bei Fehlern, die nicht vom Anwender verursacht wurden, wird die Fehlerbehandlungsroutine nicht ausgeführt.

Fehler werden eingeteilt in behebbare, nicht behebbare und nicht vom Anwender verursachte Fehler. Eine vollständige Liste aller Fehler finden Sie unter „Fehlermeldungen und Fehlercodes“ auf Seite D-1. Die Grundtypen sind nachfolgend beschrieben:

| Behebbar | Nicht behebbar | Nicht vom Anwender verursachter Fehler |
|---|---|---|
| Behebbare Fehler sind durch den Anwender verursacht und können durch Ausführen von Logik in der Anwender-fehlerroutine behoben werden. Der Anwender kann versuchen, das Bit Halt wegen schwerem Fehlers, S:1/13, zu löschen.<br>**Hinweis:** Zur Bestimmung des Fehlerzustands der Steuerung können Sie von der Steuerung aus einen MSG-Befehl an ein anderes Gerät starten. | Nicht behebbare Fehler werden durch den Anwender verursacht und können nicht behoben werden. Tritt ein derartiger Fehler auf, wird die Fehlerbehandlungsroutine ausgeführt. Der Fehler kann jedoch nicht beseitigt werden.<br>**Hinweis:** Zur Bestimmung des Fehlerzustands der Steuerung können Sie einen MSG-Befehl an ein anderes Gerät starten. | Nicht vom Anwender verursachte Fehler werden von diversen Bedingungen verursacht, die die Aus-führung des Kontaktplans zum Stillstand bringen. Tritt ein derartiger Fehler auf, wird die Fehlerbehand-lungsroutine nicht ausgeführt. |

### *Sicherung von Statusfile-Daten*

Die Arithmetik-Flags (Wort S:0 des Statusfiles) werden bei Beginn des Fehlerbehandlungs-Unterprogramms gesichert und bei Verlassen des Unterprogramms erneut geschrieben.

### *Anwenderfehler-Unterprogramm erstellen*

Gehen Sie wie folgt vor, um eine Fehlerbehandlungsroutine zu erstellen:

1. Erstellen Sie einen Unterprogrammfile. Die Programmfiles 3 bis 255 können verwendet werden.
2. Geben Sie die Filenummer in Wort S:29 des Statusfiles ein.

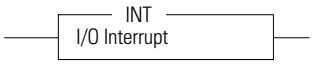
### *Betrieb der Steuerung*

Bei Eintreten eines behebbaren oder nicht behebbaren Fehlers liest die Steuerung S:29 und führt das Unterprogramm aus, das durch S:29 bezeichnet wird. Bei behebbaren Fehlern kann das Problem mit Hilfe der Routine behoben und das Fehler-Bit S:1/13 gelöscht werden. Daraufhin wird der Betrieb der Steuerung im aktuellen Ausführungs- modus fortgesetzt. Bei Fehlern, die nicht vom Anwender verursacht wurden, wird diese Routine nicht ausgeführt.

## Anwender-Interrupt-Befehle

| Befehl | Zweck | Seite |
|---|---|---|
| INT - Interrupt-Unterprogramm | Verwenden Sie diesen Befehl, um einen Programmfile als Interrupt-Unterprogramm (INT-Marke) gegenüber einem herkömmlichen Unterprogramm (SBR-Marke) zu kennzeichnen. Dies sollte der erste Befehl in einem Interrupt-Unterprogramm sein. | 18-7 |
| STS - STI starten | Verwenden Sie den STS-Befehl (STI starten), um das STI-Zeitwerk nicht automatisch, sondern über das Steuerprogramm zu starten. | 18-8 |
| UID - Anwender-Interrupt deaktivieren | Verwenden Sie die Befehle UID (Anwender-Interrupt deaktivieren) und UIE (Anwender-Interrupt aktivieren), um Zonen zu erstellen, in denen keine E/A-Interrupts auftreten können. | 18-9 |
| UIE - Anwender-Interrupt aktivieren | | 18-10 |
| UIF - Anwender-Interrupt entfernen | Verwenden Sie den UIF-Befehl, um ausgewählte, anstehende Interrupts zu entfernen. | 18-11 |

## INT – Interrupt-Unterprogramm




Befehlstyp: Eingang

**Tabelle 18.1 Ausführungszeit des INT-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 1,0 µs | 1,0 µs |
| MicroLogix 1500 | 1,0 µs | 1,0 µs |

Der INT-Befehl wird als Marke zur Kennzeichnung einer Anwender-Interrupt-Routine (ISR) verwendet. Dieser Befehl ist der erste Befehl auf einem Strompfad und wird immer als wahr bewertet. Die Verwendung des INT-Befehls ist nicht zwingend erforderlich.

# STS – STI starten

Befehlstyp: Ausgang

**Tabelle 18.2 Ausführungszeit des STS-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 57,5 µs | 0,0 µs |
| MicroLogix 1500 | 50,7 µs | 0,0 µs |

Mit dem STS-Befehl kann die STI-Funktion gestartet und gestoppt oder das Zeitintervall zwischen STI-Anwender-Interrupts geändert werden. Der STI-Befehl wird mit einem Operanden eingegeben:

- **Zeit** – Dies ist die festgelegte Zeitspanne (in Millisekunden) bis zur Ausführung des wählbar zeitgesteuerten Anwender-Interrupts. Geben Sie den Wert 0 ein, um die STI-Funktion zu deaktivieren. Gültiger Bereich: 0 bis 65.535 Millisekunden.

Der STS-Befehl wendet den angegebenen Sollwert wie folgt auf die STI-Funktion an:

- Wenn der Sollwert 0 angegeben wurde, werden die STI-Funktion deaktiviert und STI:0/TIE gelöscht (0).
- Wenn die STI-Funktion deaktiviert ist (keine Zeitsteuerung) und ein Wert größer 0 als Sollwert angegeben wurde, werden die STI-Funktion mit dem neuen Sollwert gestartet und STI:0/TIE gesetzt (1).
- Wenn die STI-Funktion bereits gestartet wurde und der Sollwert geändert wird, tritt die neue Einstellung sofort in Kraft, und die Ausführung der STI-Funktion wird mit dem neuen Sollwert fortgesetzt.

  Beachten Sie, dass bei Eingabe eines neuen Sollwerts, der kleiner als der aktuelle Istwert ist, die STI-Funktion sofort beendet wird. Wenn beispielsweise der aktuelle Istwert der STI-Funktion 15 Mikrosekunden beträgt und der STI-Sollwert von 20 Mikro- sekunden auf 10 Mikrosekunden reduziert wird, tritt beim nächsten Strompfadbeginn ein STI-Anwender-Interrupt ein.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 18.3 Gültige Adressierungsmodi und Filetpyen für STS-Befehl**

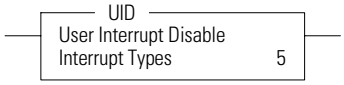
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus[1] | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Element | Bit | Wort | Doppelwort |
| Zeit | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | • | • | | | • | |

(1) Bitte beachten Sie den nachfolgenden Hinweis („Wichtig") zur indirekten Adressierung.

**WICHTIG** Bei folgenden Dateien kann die indirekte Adressierung nicht verwendet werden: S-, ST-, MG-, PD-, RTC-, HSC-, PTO-, PWM-, STI-, EII-, BHI-, MMI-, DAT-, TPI-, CS-, IOS- und DLS-Files.

# UID – Anwender-Interrupt deaktivieren




Befehlstyp: Ausgang

**Tabelle 18.4 Ausführungszeit des UID-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 0,8 µs | 0,0 µs |
| MicroLogix 1500 | 0,8 µs | 0,0 µs |

Mit dem UID-Befehl werden ausgewählte Anwender-Interrupts deaktiviert. Die nachfolgende Tabelle zeigt die verschiedenen Interrupt-Typen und die entsprechenden Deaktivierungs-Bits:

**Tabelle 18.5 Interrupt-Typen, die durch den UID-Befehl deaktiviert werden**

| Interrupt | Element | Dezimalwert | Entsprechendes Bit |
|---|---|---|---|
| EII - Ereigniseingangs-Interrupts | Ereignis 0 | 64 | Bit 6 |
| EII - Ereigniseingangs-Interrupts | Ereignis 1 | 32 | Bit 5 |
| HSC - Hochgeschwindigkeitszähler | HSC0 | 16 | Bit 4 |
| EII - Ereigniseingangs-Interrupts | Ereignis 2 | 8 | Bit 3 |
| EII - Ereigniseingangs-Interrupts | Ereignis 3 | 4 | Bit 2 |
| HSC - Hochgeschwindigkeitszähler(1) | HSC1 | 2 | Bit 1 |
| STI - Wählbar zeitgesteuerte Interrupts | STI | 1 | Bit 0 |
| Hinweis: Bits 7 bis 15 müssen auf null gesetzt werden. | | | |

(1) Die MicroLogix 1200-Steuerung verfügt über einen HSC-Interrupt, HSC0. Die MicroLogix 1500-Steuerung verfügt über zwei HSC-Interrupts, HSC0 und HSC1.

Interrupts deaktivieren:

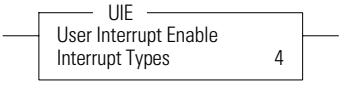
1. Wählen Sie die Interrupts aus, die deaktiviert werden sollen.
2. Ermitteln Sie den Dezimalwert des ausgewählten Interrupts.
3. Addieren Sie die Dezimalwerte, wenn Sie mehrere Interrupt-Typen ausgewählt haben.
4. Geben Sie die Summe mit dem UID-Befehl ein.

Beispiel: Deaktivierung von EII Ereignis 1 und EII Ereignis 3:

EII Ereignis 1 = 32, EII Ereignis 3 = 4

32 + 4 = 36 (diesen Wert eingeben)

# UIE – Anwender-Interrupt aktivieren




Befehlstyp: Ausgang

**Tabelle 18.6 Ausführungszeit des UIE-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 0,8 µs | 0,0 µs |
| MicroLogix 1500 | 0,8 µs | 0,0 µs |

Mit dem UIE-Befehl werden ausgewählte Anwender-Interrupts aktiviert. Die nachfolgende Tabelle zeigt die verschiedenen Interrupt- Typen und die entsprechenden Aktivierungs-Bits:

**Tabelle 18.7 Interrupt-Typen, die durch den UIE-Befehl aktiviert werden**

| Interrupt | Element | Dezimalwert | Entsprechendes Bit |
|---|---|---|---|
| EII - Ereigniseingangs-Interrupts | Ereignis 0 | 64 | Bit 6 |
| EII - Ereigniseingangs-Interrupts | Ereignis 1 | 32 | Bit 5 |
| HSC - Hochgeschwindigkeitszähler | HSC0 | 16 | Bit 4 |
| EII - Ereigniseingangs-Interrupts | Ereignis 2 | 8 | Bit 3 |
| EII - Ereigniseingangs-Interrupts | Ereignis 3 | 4 | Bit 2 |
| HSC - Hochgeschwindigkeitszähler(1) | HSC1 | 2 | Bit 1 |
| STI - Wählbar zeitgesteuerte Interrupts | STI | 1 | Bit 0 |
| Hinweis: Bits 7 bis 15 müssen auf null gesetzt werden. | | | |

(1) Die MicroLogix 1200-Steuerung verfügt über einen HSC-Interrupt, HSC0. Die MicroLogix 1500-Steuerung verfügt über zwei HSC-Interrupts, HSC0 und HSC1.

Interrupts aktivieren:

1. Wählen Sie die Interrupts aus, die aktiviert werden sollen.
2. Ermitteln Sie den Dezimalwert des ausgewählten Interrupts.
3. Addieren Sie die Dezimalwerte, wenn Sie mehrere Interrupt-Typen ausgewählt haben.
4. Geben Sie die Summe mit dem UIE-Befehl ein.

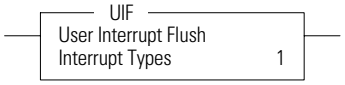
Beispiel: Aktivierung von EII Ereignis 1 und EII Ereignis 3:

EII-Ereignis 1 = 32, EII-Ereignis 3 = 4
32 + 4 = 36 (geben Sie diesen Wert ein)

**ACHTUNG**

Wenn Interrupts während der Programmabfrage über einen OTL-, OTE- oder UIE-Befehl aktiviert werden, *muss* dieser Befehl der *letzte* Befehl sein, der auf dem Strompfad ausgeführt wird (letzter Befehl auf letztem Abzweig). Es wird empfohlen, in dem Strompfad diesen Befehl als einzigen Ausgangsbefehl zu verwenden.

# UIF – Anwender-Interrupt entfernen




Befehlstyp: Ausgang

**Tabelle 18.8 Ausführungszeit für den UIF-Befehl**

| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 12,3 µs | 0,0 µs |
| MicroLogix 1500 | 10,6 µs | 0,0 µs |

Mit dem UIF-Befehl werden ausgewählte anstehende Anwender- Interrupts aus dem System entfernt. Die nachfolgende Tabelle zeigt die verschiedenen Interrupt-Typen und die entsprechenden Entfernen-Bits:

**Tabelle 18.9 Interrupt-Typen, die durch den UIF-Befehl deaktiviert werden**

| Interrupt | Element | Dezimalwert | Entsprechendes Bit |
|---|---|---|---|
| EII - Ereigniseingangs-Interrupts | Ereignis 0 | 64 | Bit 6 |
| EII - Ereigniseingangs-Interrupts | Ereignis 1 | 32 | Bit 5 |
| HSC - Hochgeschwindigkeitszähler | HSC0 | 16 | Bit 4 |
| EII - Ereigniseingangs-Interrupts | Ereignis 2 | 8 | Bit 3 |
| EII - Ereigniseingangs-Interrupts | Ereignis 3 | 4 | Bit 2 |
| HSC - Hochgeschwindigkeitszähler(1) | HSC1 | 2 | Bit 1 |
| STI - Wählbar zeitgesteuerte Interrupts | STI | 1 | Bit 0 |
| Hinweis: Bits 7 bis 15 müssen auf null gesetzt werden. | | | |

(1) Die MicroLogix 1200-Steuerung verfügt über einen HSC-Interrupt, HSC0. Die MicroLogix 1500-Steuerung verfügt über zwei HSC-Interrupts, HSC0 und HSC1.

Interrupts entfernen:

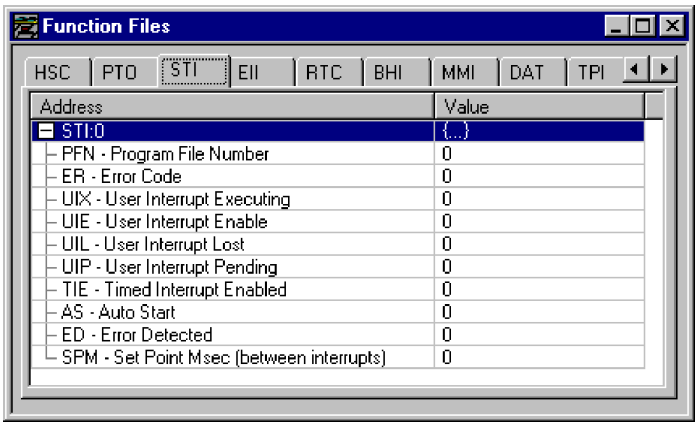
1. Wählen Sie die Interrupts aus, die entfernt werden sollen.
2. Ermitteln Sie den Dezimalwert des ausgewählten Interrupts.
3. Addieren Sie die Dezimalwerte, wenn Sie mehrere Interrupt-Typen ausgewählt haben.
4. Geben Sie die Summe mit dem UIF-Befehl ein.

Beispiel: Deaktivierung von EII Ereignis 1 und EII Ereignis 3:

EII Ereignis 1 = 32, EII Ereignis 3 = 4

32 + 4 = 36 (diesen Wert eingeben)

# STI-Funktionsfile verwenden




Mit dem wählbar zeitgesteuerten Interrupt (STI) können zeitkritische Steuerungsanforderungen gelöst werden. Der STI ist ein Auslösemechanismus zur Abfrage oder Auflösung zeitgesteuerter Steuerprogrammlogik.

Anwendungsbeispiele für STI:

- PID-Anwendungen, bei denen in bestimmten Zeitabständen eine Berechnung durchgeführt werden muss.
- Bewegungsanwendungen, bei denen Bewegungsbefehle (PTO) in bestimmten Abständen abgefragt werden müssen, um ein gleichmäßiges Beschleunigungs-/Verzögerungsprofil sicherzustellen.
- Logikblocks, die mehrfach abgefragt werden müssen.

Die genaue Verwendung eines STI hängt in der Regel von den Anforderungen der konkreten Anwendung ab. STI werden in folgender Reihenfolge ausgeführt:

1. Der Anwender wählt ein Zeitintervall aus.
2. Nach Eingabe eines gültigen Zeitintervalls und korrekter Konfiguration des STI wird der STI-Wert durch die Steuerung überwacht.
3. Nach Ablauf der festgelegten Zeitspanne wird der normale Steuerungsbetrieb unterbrochen.
4. Die Logik in dem STI-Programmfile wird abgefragt.
5. Nach Abschluss der Abfrage des STI-Files wird der normale Steuerungsbetrieb an der Stelle fortgesetzt, an der der Interrupt auftrat.

## Zusammenfassung der Unterelemente des STI-Funktionsfiles

**Tabelle 18.10 STI-Funktionsfile (STI:0)**

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff | Weitere Informationen |
|---|---|---|---|---|---|
| PFN - Programmfilenummer | STI:0.PFN | Wort (INT) | Steuerung | Nur Lesen | 18-13 |
| ER - Fehlercode | STI:0.ER | Wort (INT) | Status | Nur Lesen | 18-13 |
| UIX - Anwender-Interrupt-Ausführung | STI:0/UIX | Binärwert (Bit) | Status | Nur Lesen | 18-14 |
| UIE - Anwender-Interrupt aktivieren | STI:0/UIE | Binärwert (Bit) | Steuerung | Lesen/Schreiben | 18-14 |
| UIL - Anwender-Interrupt-Verlust | STI:0/UIL | Binärwert (Bit) | Status | Lesen/Schreiben | 18-14 |
| UIP - Anwender-Interrupt anstehend | STI:0/UIP | Binärwert (Bit) | Status | Nur Lesen | 18-15 |
| TIE - Zeitgesteuerter Interrupt aktiviert | STI:0/TIE | Binärwert (Bit) | Steuerung | Lesen/Schreiben | 18-15 |
| AS - Auto-Start | STI:0/AS | Binärwert (Bit) | Steuerung | Nur Lesen | 18-15 |
| ED - Fehler erkannt | STI:0/ED | Binärwert (Bit) | Status | Nur Lesen | 18-16 |
| SPM - Sollwert | STI:0.SPM | Wort (INT) | Steuerung | Lesen/Schreiben | 18-16 |

## Unterelemente des STI-Funktionsfiles

*STI-Programmfilenummer (PFN)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| PFN - Programmfilenummer | STI:0.PFN | Wort (INT) | Steuerung | Nur Lesen |

Die PFN-Variable (Programmfilenummer) legt das Unterprogramm fest, das aufgerufen (ausgeführt) wird, wenn der zeitgesteuerte Interrupt eintritt. Als Unterprogrammfile kann jeder Programmfile (3 bis 255) festgelegt werden.

Der Unterprogrammfile, der durch die PFN-Variable angegeben wird, ist kein spezieller File innerhalb der Steuerung. Er wird vielmehr wie jeder andere Programmfile programmiert und eingesetzt. Aus Sicht des Steuerprogramms liegt die Besonderheit dieses Files darin, dass der File entsprechend dem STI-Sollwert automatisch abgefragt wird.

*STI-Fehlercode (ER)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| ER - Fehlercode | STI:0.ER | Wort (INT) | Status | Nur Lesen |

Fehlercodes, die von dem STI-Subsystem erkannt wurden, werden in diesem Register angezeigt. Die Fehlercodes werden in der nachfolgen- den Tabelle beschrieben.

**Tabelle 18.11 STI-Fehlercode**

| Fehlercode | Behebbarer Fehler (Steuerung) | Beschreibung |
|---|---|---|
| 1 | Ungültige Programmfilenummer | Programmfilenummer kleiner als 3, größer als 255 oder nicht vorhanden |

*STI-Anwender-Interrupt-Ausführung (UIX)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| UIX - Anwender-Interrupt-Ausführung | STI:0/UIX | Binärwert (Bit) | Status | Nur Lesen |

Das UIX-Bit (Anwender-Interrupt-Ausführung) wird gesetzt, sobald die STI-Funktion die Zeitmessung abgeschlossen hat und die STI-PFN durch die Steuerung abgefragt wird. Das UIX-Bit wird gelöscht, wenn die Verarbeitung des STI-Unterprogramms durch die Steuerung abgeschlossen ist.

Das UIX-Bit kann in dem Steuerprogramm als bedingte Logik eingesetzt werden, um festzustellen, wenn ein STI-Interrupt ausgeführt wird.

*STI-Anwender-Interrupt aktivieren (UIE)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| UIE - Anwender-Interrupt aktivieren | STI:0/UIE | Binärwert (Bit) | Steuerung | Lesen/Schreiben |

Mit dem UIE-Bit (Anwender-Interrupt aktivieren) wird die Verar- beitung von STI-Unterprogrammen aktiviert oder deaktiviert. Dieses Bit muss gesetzt werden, wenn das STI-Unterprogramm in den festgelegten Zeitintervallen durch die Steuerung verarbeitet werden soll.

Löschen Sie das UIE-Bit, wenn die Ausführung des STI- Unter- programms zeitlich eingeschränkt werden soll. Dies könnte beispielsweise erforderlich sein, wenn eine Reihe mathematischer Berechnungen ohne Unterbrechung ausgeführt werden sollen. Löschen Sie das UIE-Bit vor der Durchführung der Berechnungen. Setzen Sie das UIE-Bit nach Abschluss der Berechnungen; die Verarbeitung des STI-Unterprogramms wird daraufhin fortgesetzt.

*STI-Anwender-Interrupt-Verlust (UIL)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| UIL - Anwender-Interrupt-Verlust | STI:0/UIL | Binärwert (Bit) | Status | Lesen/Schreiben |

Das UIL-Status-Flag (Anwender-Interrupt-Verlust) zeigt an, dass ein Interrupt verloren wurde. Die Steuerung kann ein aktives und bis zu zwei anstehende Anwender-Interrupt-Bedingungen verarbeiten, bevor das UIL-Bit gesetzt wird.

Dieses Bit wird von der Steuerung gesetzt. Die Verlustbedingung muss durch das Steuerprogramm verwendet, ggf. verfolgt und gelöscht werden.

### *STI-Anwender-Interrupt anstehend (UIP)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| UIP - Anwender-Interrupt anstehend | STI:0/UIP | Binärwert (Bit) | Status | Nur Lesen |

Das UIP-Status-Flag (Anwender-Interrupt anstehend) zeigt an, dass ein Interrupt ansteht. Dieses Status-Bit kann überwacht oder als Logik in dem Steuerprogramm eingesetzt werden, um festzustellen, wenn ein Unterprogramm nicht sofort ausgeführt werden kann.

Dieses Bit wird durch die Steuerung automatisch gesetzt und gelöscht. Die Steuerung kann ein aktives und bis zu zwei anstehende Anwender-Interrupt-Bedingungen verarbeiten, bevor das UIL-Bit gesetzt wird.

### *STI Zeitgesteuerter Interrupt aktiviert (TIE)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| TIE - Zeitgesteuerter Interrupt aktiviert | STI:0/TIE | Binärwert (Bit) | Steuerung | Lesen/Schreiben |

Mit dem TIE-Steuer-Bit (Zeitgesteuerter Interrupt aktiviert) werden zeitgesteuerte Interrupts aktiviert oder deaktiviert. Durch Setzen dieses Bits (1) wird die Zeitmessung aktiviert; wenn das Bit gelöscht ist (0), ist die Zeitmessung deaktiviert. Wenn dieses Bit gelöscht wird, während die Zeitmessung läuft, wird der Istwert gelöscht (0). Wenn das Bit anschließend gesetzt wird (1), beginnt die Zeitmessung.

Dieses Bit wird durch das Anwenderprogramm gesteuert; der Wert dieses Bits bleibt auch nach Aus- und Einschalten der Spannungs- versorgung erhalten.

### *STI Auto-Start (AS)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| AS - Auto-Start | STI:0/AS | Binärwert (Bit) | Steuerung | Nur Lesen |

Das AS-Bit (Auto-Start) ist ein Steuer-Bit, das in dem Steuerprogramm verwendet werden kann. Das Auto-Start-Bit wird über das Program- miergerät konfiguriert und als Teil des Anwenderprogramms gespeichert. Das Auto-Start-Bit setzt automatisch das TIE-Bit, wenn die Steuerung in einen Ausführungsmodus schaltet.

*STI-Fehler erkannt (ED)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| ED - Fehler erkannt | STI:0/ED | Binärwert (Bit) | Status | Nur Lesen |

Das ED-Flag (Fehler erkannt) ist ein Status-Bit, das in dem Steuer- programm verwendet werden kann, um Fehler in dem STI-Subsystem festzustellen. Die häufigsten Fehler, die dieses Bit darstellt, sind Konfigurationsfehler. Wenn dieses Bit gesetzt ist, sollte der spezifische Fehlercode in dem Parameter STI:0.ER genauer geprüft werden.

Dieses Bit wird durch die Steuerung automatisch gesetzt und gelöscht.

*STI Sollwert (SPM, Millisekunden zwischen Interrupts)*

| Beschreibung Unterelement | Adresse | Datenformat | Bereich | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|---|
| SPM - Sollwert | STI:0.SPM | Wort (INT) | 0 bis 65535 | Steuerung | Lesen/Schreiben |

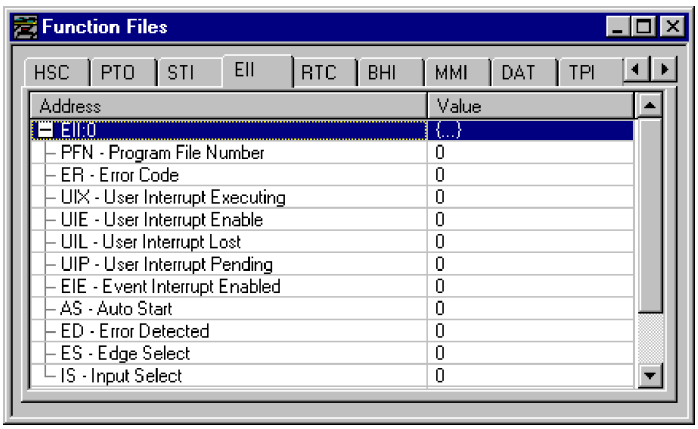
Beim Übergang der Steuerung in den Ausführungsmodus wird der SPM-Wert (Sollwert in Millisekunden) in das STI-Subsystem geladen. Wenn das STI-Subsystem korrekt konfiguriert und aktiviert wurde, wird der in der STI-Variablen SPN genannte Programmfile in diesen Intervallen abgefragt. Der Wert kann in dem Steuerprogramm mit Hilfe des STS-Befehls geändert werden.

**HINWEIS** Dieser Wert darf nicht kleiner sein als die erforderliche Abfragezeit des STI-Programmfiles (STI:0.PFN) plus der Interrupt-Latenzzeit.

## EII-Funktionsfile verwenden

Der Ereigniseingangs-Interrupt (EII) ist eine Funktion, die die Abfrage eines bestimmten Programmfiles (Unterprogramm) ermöglicht, wenn eine Eingangsbedingung an einem Feldgerät festgestellt wird.

Unter den Funktionsfiles der RSLogix 500 befindet sich auch der Ordner EII. Dieser Ordner enthält vier EII-Elemente. Diese Elemente (EII:0, EII:1, EII:2 und EII:3) sind identisch; in der nachfolgenden Beschreibung wird EII:0 als Beispiel verwendet.




Jeder EII kann für die Überwachung eines der ersten acht Eingänge (I1:0.0/0 bis I1:0.0/7) konfiguriert werden. Außerdem kann jeder EII für die Erkennung von Anstiegsflanken- oder abfallende Flanken- Eingangssignalen konfiguriert werden. Wenn das ausgewählte Eingangssignal an der Eingangsklemme erkannt wird, führt die Steuerung sofort eine Abfrage des konfigurierten Unterprogramms durch.

## Zusammenfassung der Unterelemente des EII-Funktionsfiles (Ereigniseingangs-Interrupt)

**Tabelle 18.12 EII-Funktionsfile (Ereigniseingangs-Interrupt, EII:0)**

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff | Weitere Informationen |
|---|---|---|---|---|---|
| PFN - Programmfilenummer | EII:0.PFN | Wort (INT) | Steuerung | Nur Lesen | 18-18 |
| ER - Fehlercode | EII:0.ER | Wort (INT) | Status | Nur Lesen | 18-18 |
| UIX - Anwender-Interrupt-Ausführung | EII:0/UIX | Binärwert (Bit) | Status | Nur Lesen | 18-18 |
| UIE - Anwender-Interrupt aktivieren | EII:0/UIE | Binärwert (Bit) | Steuerung | Lesen/Schreiben | 18-19 |
| UIL - Anwender-Interrupt-Verlust | EII:0/UIL | Binärwert (Bit) | Status | Lesen/Schreiben | 18-19 |
| UIP - Anwender-Interrupt anstehend | EII:0/UIP | Binärwert (Bit) | Status | Nur Lesen | 18-19 |
| EIE - Ereignis-Interrupt aktiviert | EII:0/EIE | Binärwert (Bit) | Steuerung | Lesen/Schreiben | 18-20 |
| AS - Auto-Start | EII:0/AS | Binärwert (Bit) | Steuerung | Nur Lesen | 18-20 |
| ED - Fehler erkannt | EII:0/ED | Binärwert (Bit) | Status | Nur Lesen | 18-20 |
| ES - Flankenauswahl | EII:0/ES | Binärwert (Bit) | Steuerung | Nur Lesen | 18-21 |
| IS - Eingangsauswahl | EII:0.IS | Wort (INT) | Steuerung | Nur Lesen | 18-21 |

## Unterelemente des EII-Funktionsfiles

*EII-Programmfilenummer (PFN)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| PFN - Programmfilenummer | EII:0.PFN | Wort (INT) | Steuerung | Nur Lesen |

Die PFN-Variable (Programmfilenummer) legt das Unterprogramm fest, das aufgerufen (ausgeführt) wird, wenn die Eingangsklemme, die EII:0 zugeordnet ist, ein Signal erkennt. Als Unterprogrammfile kann jeder Programmfile (3 bis 255) festgelegt werden.

Der Unterprogrammfile, der durch die PFN-Variable angegeben wird, ist kein spezieller File innerhalb der Steuerung. Er wird vielmehr wie jeder andere Programmfile programmiert und eingesetzt. Aus Sicht des Steuerprogramms liegt die Besonderheit dieses Files darin, dass der File entsprechend der EII-Konfiguration automatisch abgetastet wird.

## EII-Fehlercode (ER)

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| ER - Fehlercode | EII:0.ER | Wort (INT) | Status | Nur Lesen |

Fehlercodes, die von dem EII-Subsystem erkannt wurden, werden in diesem Register angezeigt. Die Fehlercodes werden in der nachfolgen- den Tabelle beschrieben.

**Tabelle 18.13 EII-Fehlercodes**

| Fehler-code | Behebbarer Fehler (Steuerung) | Beschreibung |
|---|---|---|
| 1 | Ungültige Programmfilenummer | Programmfilenummer kleiner als 3, größer als 255 oder nicht vorhanden |
| 2 | Ungültige Eingangsauswahl | Gültige Nummern sind 0, 1, 2, 3, 4, 5, 6 oder 7 |
| 3 | Überschneidung bei Eingangsauswahl | Eine gemeinsame Benutzung eines Eingangs durch mehrere EII ist nicht möglich. Jedem EII muss ein separater Eingang zugeordnet werden. |

*EII Anwender-Interrupt-Ausführung (UIX)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| UIX - Anwender-Interrupt-Ausführung | EII:0/UIX | Binärwert (Bit) | Status | Nur Lesen |

Das UIX-Bit (Anwender-Interrupt-Ausführung) wird gesetzt, sobald die EII-Funktion einen gültigen Eingang erkennt und die PFN durch die Steuerung abgefragt wird. Das EII-Subsystem löscht das UIX-Bit, wenn die Verarbeitung des EII-Unterprogramms durch die Steuerung abgeschlossen ist.

Das UIX-Bit kann in dem Steuerprogramm als bedingte Logik eingesetzt werden, um festzustellen, ob ein EII-Interrupt ausgeführt wird.

### *EII Anwender-Interrupt aktivieren (UIE)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| UIE - Anwender-Interrupt aktivieren | EII:0/UIE | Binärwert (Bit) | Steuerung | Lesen/Schreiben |

Mit dem UIE-Bit (Anwender-Interrupt aktivieren) wird die Verar- beitung von EII-Unterprogrammen aktiviert oder deaktiviert. Dieses Bit muss gesetzt werden, wenn das EII-Unterprogramm bei Eintritt eines EII-Ereignisses verarbeitet werden soll.

Löschen Sie das UIE-Bit, wenn die Ausführung des EII-Unter- programms zeitlich eingeschränkt werden soll. Dies könnte beispielsweise erforderlich sein, wenn eine Reihe mathematischer Berechnungen ohne Unterbrechung ausgeführt werden sollen. Löschen Sie das UIE-Bit vor der Durchführung der Berechnungen. Setzen Sie das UIE-Bit nach Abschluss der Berechnungen; die Verarbeitung des EII-Unterprogramms wird daraufhin fortgesetzt.

### *EII Anwender-Interrupt-Verlust (UIL)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| UIL - Anwender-Interrupt-Verlust | EII:0/UIL | Binärwert (Bit) | Status | Lesen/Schreiben |

Das UIL-Status-Flag (Anwender-Interrupt-Verlust) zeigt an, dass ein Interrupt verloren wurde. Die Steuerung kann ein aktives und bis zu zwei anstehende Anwender-Interrupt-Bedingungen verarbeiten, bevor das UIL-Bit gesetzt wird.

Dieses Bit wird von der Steuerung gesetzt. Die Verlustbedingung muss durch das Steuerprogramm verwendet, ggf. verfolgt und gelöscht werden.

### *EII Anwender-Interrupt anstehend (UIP)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| UIP - Anwender-Interrupt anstehend | EII:0/UIP | Binärwert (Bit) | Status | Nur Lesen |

Das UIP-Status-Flag (Anwender-Interrupt anstehend) zeigt an, dass ein Interrupt ansteht. Dieses Status-Bit kann überwacht oder als Logik in dem Steuerprogramm eingesetzt werden, um festzustellen, wenn ein Unterprogramm nicht sofort ausgeführt werden kann.

Dieses Bit wird durch die Steuerung automatisch gesetzt und gelöscht. Die Steuerung kann ein aktives und bis zu zwei anstehende Anwender-Interrupt-Bedingungen verarbeiten, bevor das UIP-Bit gesetzt wird.

### *EII Ereignis-Interrupt aktiviert (EIE)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| EIE - Ereignis-Interrupt aktiviert | EII:0/EIE | Binärwert (Bit) | Steuerung | Lesen/Schreiben |

Mit dem EIE-Bit (Ereignis-Interrupt aktiviert) können Sie die Ereignis-Interrupt-Funktion über das Steuerprogramm aktivieren oder deaktivieren. Wenn das Bit gesetzt ist (1), ist die Funktion aktiviert; bei gelöschtem Bit (0, Standard) ist die Funktion deaktiviert.

Dieses Bit wird durch das Anwenderprogramm gesteuert; der Wert dieses Bits bleibt auch nach Aus- und Einschalten der Spannungs- versorgung erhalten.

### *EII Auto-Start (AS)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| AS - Auto-Start | EII:0/AS | Binärwert (Bit) | Steuerung | Nur Lesen |

Das AS-Bit (Auto-Start) ist ein Steuer-Bit, das in dem Steuerprogramm verwendet werden kann. Das Auto-Start-Bit wird über das Program- miergerät konfiguriert und als Teil des Anwenderprogramms gespeichert. Das Auto-Start-Bit setzt automatisch das EIE-Bit, wenn die Steuerung in einen Ausführungsmodus schaltet.

### *EII Fehler erkannt (ED)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| ED - Fehler erkannt | EII:0/ED | Binärwert (Bit) | Status | Nur Lesen |

Das ED-Flag (Fehler erkannt) ist ein Status-Bit, das in dem Steuer- programm verwendet werden kann, um Fehler in dem EII-Subsystem festzustellen. Die häufigsten Fehler, die dieses Bit darstellt, sind Konfigurationsfehler. Wenn dieses Bit gesetzt ist, sollte der spezifische Fehlercode in dem Parameter EII:0.ER genauer geprüft werden.

Dieses Bit wird durch die Steuerung automatisch gesetzt und gelöscht.

### *EII Flankenauswahl (ES)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| ES - Flankenauswahl | EII:0/ES | Binärwert (Bit) | Steuerung | Nur Lesen |

Mit dem ES-Bit (Flankenauswahl) wird der Auslösertyp ausgewählt, der einen Ereignis-Interrupt verursacht. Dabei kann das EII-Subsystem für die Erkennung eines Anstiegsflankensignals (aus-ein, 0-1) oder eines abfallende Flankensignals (ein-aus, 1-0) konfiguriert werden. Diese Auswahl hängt von dem Feldgerätetyp ab, das an die Steuerung angeschlossen ist.

Die Standardbedingung ist 1, Anstiegsflanke.

### *EII Eingangsauswahl (IS)*

| Beschreibung Unterelement | Adresse | Datenformat | Typ | Anwenderpro-grammzugriff |
|---|---|---|---|---|
| IS - Eingangsauswahl | EII:0.IS | Wort (INT) | Steuerung | Nur Lesen |

Mit dem IS-Parameter (Eingangsauswahl) wird jedem EII ein bestimm- ter Eingang auf der Steuerung zugeordnet. Gültige Eingänge sind 0 bis 7; dies entspricht I1:0.0/0 bis I1:0.0/7.

Dieser Parameter wird über das Programmiergerät konfiguriert und kann nicht in dem Steuerprogramm geändert werden.

# Prozesssteuerungsbefehl

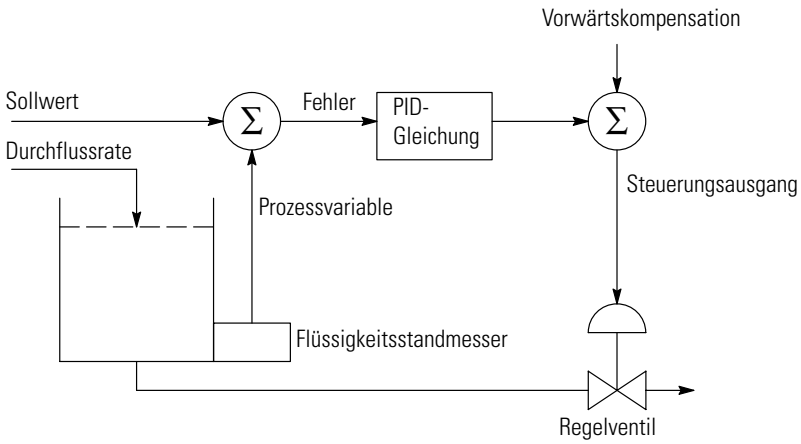
In diesem Kapitel wird der PID-Befehl (Proportional Integral Derivative) für MicroLogix 1200- und MicroLogix 1500-Steuerungen beschrieben. Der PID-Befehl ist ein Ausgangsbefehl, der physikalische Parameter wie Temperatur, Druck, Flüssigkeitsstand oder die Strömungsgeschwindigkeit in Prozessregelkreisen regelt.

## Konzept des PID-Befehls

Normalerweise regelt der PID-Befehl einen geschlossenen Regelkreis, indem er die Eingangsdaten eines Analog-Eingangsmoduls verarbeitet und Daten an den Ausgang eines Analog-Ausgangsmoduls überträgt. Bei Temperaturregelungen können Analogausgänge in zeitabhängige Ein/ Aus-Ausgänge umgewandelt werden, um Heiz oder Kühlelemente zu regeln. Ein Beispiel hierzu finden Sie auf Seite 19-18.

Der PID-Befehl kann in einem Zeitsteuerungsmodus oder im STI- Modus (Wählbar zeitgesteuerter Interrupt) ausgeführt werden. Im Zeitsteuerungsmodus wird der Befehlsausgang in regelmäßigen, durch den Anwender festgelegten Intervallen aktualisiert. In der STI- Betriebsart sollte der Befehl in ein STI-Interrupt-Unterprogramm integriert werden. Die Ausgangsdaten werden dann bei jeder Abfrage des STI-Unterprogramms aktualisiert. Das STI-Zeitintervall und das Aktualisierungsintervall des PID-Regelkreises müssen in der gleichen Reihenfolge spezifiziert werden, damit die Gleichung korrekt abgear- beitet wird. Weitere Informationen zu STI-Interrupts finden Sie unter „STI-Funktionsfile verwenden“ auf Seite 18-12.

Durch die geschlossene PID-Regelkreisüberwachung wird der Istwert auf dem gewünschten Sollwert gehalten. Ein Beispiel für die Steue- rung einer Durchflussrate und eines Flüssigkeitsstands ist nachfolgend dargestellt.




Die PID-Gleichung regelt den Prozess durch die Übertragung eines Ausgangssignals an das Regelventil. Je größer die Abweichung zwischen dem Sollwert und dem Ausgang der Prozessvariablen, um so größer das Ausgangssignal. D. h., je kleiner die Abweichung, um so kleiner das Ausgangssignal. Zu dem Regelausgang kann ein zusätz- licher Wert

(Störgrößenaufschaltung oder Grundlast) als Versatz hinzugefügt werden. Das Ergebnis der PID-Berechnung (Stellgröße) ermöglicht die Annäherung des geregelten Istwerts an den Sollwert.

## PID-Gleichung

Der PID-Befehl enthält den folgenden Algorithmus:

**Standardgleichung mit abhängigen Proportionalverstärkungen:**

$$Output = K_C\left[(E) + \frac{1}{T_I}\int(E)dt + T_D \cdot \frac{d(PV)}{dt}\right] + bias$$

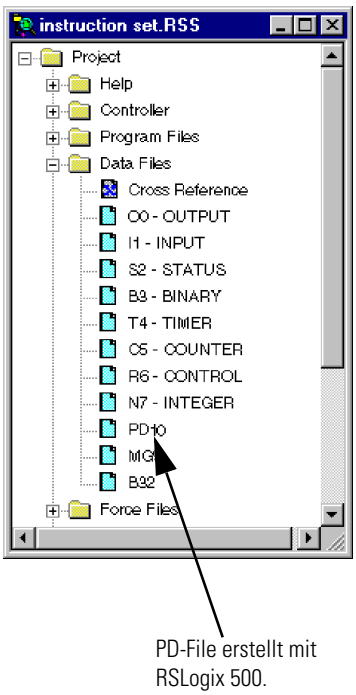
Standardmäßige Proportionalverstärkungskonstanten:

| Zeit | Bereich (Minimal und Maximalbereich) | Referenz |
|---|---|---|
| Verstärkung $K_C$ | 0,01 bis 327,67 (ohne Einheit)(1) | Proportional |
| Integralzeit $1/T_I$ | 327,67 bis 0,01 (Minuten pro Wiederholung)(1) | Integral |
| Differenzialzeit $T_D$ | 0,01 bis 327,67 (Minuten)(1) | Differenzial |

(1) Gilt für die MicroLogix 1200- und 1500 PID-Serie, wenn das RG-Bit (Reset und Gain; Zurücksetzen und Verstärkung) auf 1 gesetzt ist. Informationen zum Zurücksetzen und zur Verstärkung finden Sie im Abschnitt „PLC 5-Verstärkungsbereich (RG)“ auf Seite 19-13.

Der Differenzialterm (Differenzialanteil) ermöglicht eine Glättung mittels eines Tiefpassfilters. Die Grenzfrequenz des Filters ist 16 mal größer als die Eckfrequenz des Differenzialterms.

## PD-Datenfile



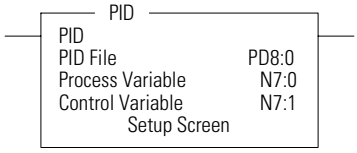

PD-File erstellt mit RSLogix 500.

Der in MicroLogix 1200 und MicroLogix 1500 verfügbare PID-Befehl ist in seiner Funktion praktisch identisch mit dem PID-Befehl, der bei den Allen-Bradley-Prozessoren SLC 5/03 und höher verwendet wird. Kleinere Unterschiede betreffen vor allem die Terminologie, die erweitert wurde. Der größte Unterschied besteht darin, dass für den PID-Befehl nun ein separater Datenfile vorhanden ist. In der SLC-Prozessorreihe wurde der PID-Befehl als Registerblock innerhalb eines Ganzzahl-Files ausgeführt. Der PID-Befehl bei Micrologix 1200 und MicroLogix 1500-Steuerungen verwendet einen PD-Datenfile.

Um einen PD-Datenfile zu erstellen, generieren Sie einen neuen Datenfile und kennzeichnen diesen dann als PD-File. RSLogix erstellt automatisch einen neuen PD-File oder ein PD-Unterelement, sobald ein PID-Befehl auf einem Strompfad programmiert wird. Der PD-File erscheint dann in der Liste der Datenfiles (siehe Abbildung).

Jeder PD-Datenfile enthält maximal 255 Elemente, und für jeden PID-Befehl ist ein separates PD-Element erforderlich. Jedes PD-Ele- ment besteht aus 20 Unterelementen, zu denen Bit-, Ganzzahl- und Doppelwortdaten gehören. Alle in diesem Kapitel beschriebenen Beispiele verwenden das Unterelement 0 des PD-Files 10.

# PID – Proportional-/ Integral-/ Differenzialverhalten




Befehlstyp: Ausgang

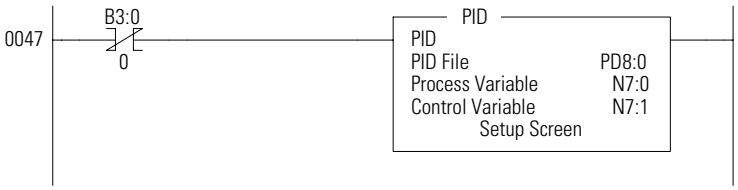
**Tabelle 19.1 Ausführungszeit des PID-Befehls**

| Steuerung | Strompfad | |
|---|---|---|
| | **wahr** | **unwahr** |
| MicroLogix 1200 | 295,8 µs | 11,0 µs |
| MicroLogix 1500 | 251,8 µs | 8,9 µs |

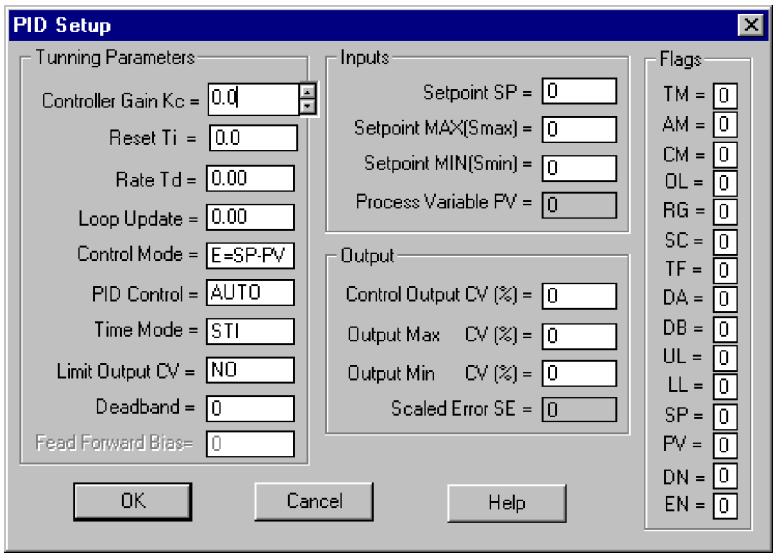
Es ist empfehlenswert, den PID-Befehl ohne bedingte Logik auf einem Strompfad zu setzen. Sofern eine bedingte Logik vorhanden ist, ändert sich der Wert des Ausgangs der Kontrollvariablen nicht, und der Aus- druck CVP CV% sowie der integrierte Ausdruck werden gelöscht, wenn der Strompfad unwahr ist.

**HINWEIS** Um einen PID-Befehl zu stoppen und erneut zu starten, muss ein Strompfadübergang von unwahr nach wahr erzeugt werden.

Das nachfolgende Beispiel zeigt einen PID-Befehl auf einem Strompfad mit einer RSLogix 500-Programmiersoftware.




Bei der Programmierung erfolgt der Zugriff auf die Konfigurations- parameter des PID-Befehls über den Einrichtungsbildschirm. Die nachfolgende Abbildung zeigt den RSLogix 500-Einrichtungs- bildschirm.



## Eingangsparameter

Die nachfolgende Tabelle zeigt die Adressen, Datenformate und Anwenderprogrammzugriffe der Eingangsparameter. Beschreibungen der einzelnen Parameter finden Sie auf den in der letzten Spalte angegebenen Seiten.

| Eingangsparameter | Adresse | Datenformat | Bereich | Typ | Anwender-programm-zugriff | Weitere Informationen |
|---|---|---|---|---|---|---|
| SPS - Sollwert | PD10:0.SPS | Wort (INT) | 0 bis 16383(1) | Steuerung | Lesen/ Schreiben | 19-4 |
| PV - Prozessvariable | anwenderdefiniert | Wort (INT) | 0 bis 16383 | Steuerung | Lesen/ Schreiben | 19-4 |
| MAXS - Max. Sollwert | PD10:0.MAXS | Wort (INT) | -32768 bis +32767 | Steuerung | Lesen/ Schreiben | 19-5 |
| MINS - Min. Sollwert | PD10:0.MINS | Wort (INT) | -32768 bis +32767 | Steuerung | Lesen/ Schreiben | 19-5 |
| OSP - Alter Sollwert | PD10:0.OSP | Wort (INT) | -32768 bis +32767 | Status | Nur Lesen | 19-5 |
| OL - Ausgangsbegrenzung | PD10:0/OL | Binärwert | 1 = aktiviert<br>0 = deaktiviert | Steuerung | Lesen/ Schreiben | 19-6 |
| CVH - Obergrenze Regelvariable | PD10:0.CVH | Wort (INT) | 0 bis 100 % | Steuerung | Lesen/ Schreiben | 19-6 |
| CVL - Untergrenze Regelvariable | PD10:0.CVL | Wort (INT) | 0 bis 100 % | Steuerung | Lesen/ Schreiben | 19-6 |

(1) Der in der Tabelle aufgeführte Bereich gilt bei nicht aktivierter Skalierung. Bei aktivierter Skalierung reicht der Bereich vom minimalen skalierten Wert (MINS) bis zum maximalen skalierten Wert (MAXS).

### Sollwert (SPS)

| Eingangsparameter | Adresse | Datenformat | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| SPS - Sollwert | PD10:0.SPS | Wort (INT) | 0 bis 16383(1) | Steuerung | Lesen/Schreiben |

(1) Der in der Tabelle aufgeführte Bereich gilt bei nicht aktivierter Skalierung. Bei aktivierter Skalierung reicht der Bereich vom minimalen skalierten Wert (MINS) bis zum maximalen skalierten Wert (MAXS).

Der SPS-Wert (Sollwert) entspricht dem gewünschten Steuerungspunkt der Prozessvariablen.

### Prozessvariable (PV)

| Eingangs-parameter | Adresse | Datenformat | Bereich | Typ | Anwender-programm-zugriff |
|---|---|---|---|---|---|
| PV - Prozessvariable | anwender definiert | Wort (INT) | 0 bis 16383 | Steuerung | Lesen/ Schreiben |

PV (Prozessvariable) entspricht der analogen Eingangsvariablen.

## Max. Sollwert (MAXS)

| Eingangs-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| MAXS - Max. Sollwert | PD10:0.MAXS | Wort (INT) | -32768 bis +32767 | Steuerung | Lesen/Schreiben |

Wenn die SPV in physikalischen Einheiten gelesen wird, entspricht der MAXS-Parameter (Max. Sollwert) dem Wert des Sollwerts in physikali- schen Einheiten bei maximalem Wert des Steuerungseingangs.

## Min. Sollwert (MINS)

| Eingangs-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| MINS - Min. Sollwert | PD10:0.MINS | Wort (INT) | -32768 bis +32767 | Steuerung | Lesen/Schreiben |

Wenn die SPV in physikalischen Einheiten gelesen wird, entspricht der MIN-Parameter (Min. Sollwert) dem Wert des Sollwerts in physikali- schen Einheiten bei minimalem Wert des Steuerungseingangs.

**HINWEIS** Bei Verwendung der *MinS – MaxS*-Skalierung können Sie in physikalischen Einheiten arbeiten. Totzone, Skalierte Abweichung und SPV werden ebenfalls in physikalischen Einheiten angezeigt. Die Prozessvari- able PV muss in einem Bereich zwischen 0 und 16383 liegen. Die PID-PV-Auflösung wird durch Verwendung der *MinS – MaxS*-Skalierung nicht beeinträchtigt.

Skalierte Abweichung über +32767 oder unter -32768 können nicht dargestellt werden. Skalierte Abweichung über +32767 werden als +32767 dargestellt. Skalierte Abweichung unter –32768 werden als –32768 dargestellt.

## Alter Sollwert (OSP)

| Eingangs-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| OSP - Alter Sollwert | PD10:0.OSP | Wort (INT) | -32768 bis +32767 | Status | Nur Lesen |

Der OSP (Alter Sollwert) ersetzt den aktuellen Sollwert, wenn der aktuelle Sollwert außerhalb des Bereichs der Sollwert-Skalier- parameter liegt.

## Ausgangsbegrenzung (OL)

| Ausgangs-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| OL - Ausgangs-begrenzung | PD10:0/OL | Binärwert | 1 = aktiviert<br>0 = deaktiviert | Steuerung | Lesen/Schreiben |

Wenn dieser Wert aktiviert (1) ist, wird der Ausgang auf die Werte begrenzt, die in PD10:0.CVH (Obergrenze Regelvariable) und PD10.0.CVL (Untergrenze Regelvariable) festgelegt sind.

Ist der Wert deaktiviert (0), erfolgt keine Ausgangsbegrenzung.

## Obergrenze Regelvariable (CVH)

| Ausgangs-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| CVH - Obergrenze Regelvariable | PD10:0.CVH | Wort (INT) | 0 bis 100 % | Steuerung | Lesen/Schreiben |

Wenn das Ausgangsbegrenzungs-Bit (PD10:0/OL) aktiviert (1) ist, stellt der eingegebene CVH-Wert (Obergrenze Regelvariable) den maxima- len Ausgang (in %) der Regelvariablen dar. Wenn die berechnete Regelvariable den CVH-Wert überschreitet, wird die Regelvariable auf den eingegebenen Wert gesetzt, und das UL-Bit (Oberer Ausgangsalarm) wird gesetzt.

Wenn das Ausgangsbegrenzungs-Bit (PD10:0/OL) deaktiviert (0) ist, wird über den eingegebenen CVH-Wert festgelegt, wann das UL-Bit (Oberer Ausgangsalarm) gesetzt wird.

Wenn die Regelvariable den maximalen Wert überschreitet, wird der Ausgang nicht überschrieben, und das UL-Bit (Oberer Ausgangsalarm) wird gesetzt.

## Untergrenze Regelvariable (CVL)

| Ausgangs-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| CVL - Untergrenze Regelvariable | PD10:0.CVL | Wort (INT) | 0 bis 100 % | Steuerung | Lesen/Schreiben |

Wenn das Ausgangsbegrenzungs-Bit (PD10:0/OL) aktiviert (1) ist, stellt der eingegebene CVL-Wert (Untergrenze Regelvariable) den Mindest- ausgang (in %) der Regelvariablen dar. Wenn die berechnete Regel- variable den CVL-Wert unterschreitet, wird die Regelvariable auf den eingegebenen Wert gesetzt, und das LL-Bit (Unterer Ausgangsalarm) wird gesetzt.

Wenn das Ausgangsbegrenzungs-Bit (PD10:0/OL) deaktiviert (0) ist, wird über den eingegebenen CVL-Wert festgelegt, wann das LL-Bit (Unterer Ausgangsalarm) gesetzt wird. Wenn die Regelvariable den Mindestwert unterschreitet, wird der Ausgang nicht überschrieben, und das LL-Bit (Unterer Ausgangsalarm) wird gesetzt.

## Ausgangsparameter

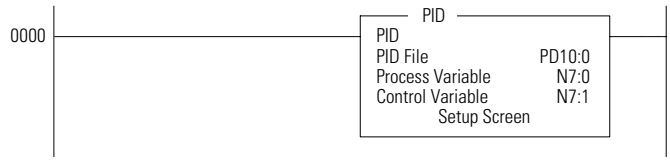
Die nachfolgende Tabelle zeigt die Adressen, Datenformate und Anwenderprogrammzugriffe der Ausgangsparameter. Beschreibungen der einzelnen Parameter finden Sie auf den in der letzten Spalte angegebenen Seiten.

| Ausgangsparameter | Adresse | Datenformat | Bereich | Typ | Anwender-programmzugriff | Weitere Informationen |
|---|---|---|---|---|---|---|
| CV - Regelvariable | anwenderdefiniert | Wort (INT) | 0 bis 16383 | Steuerung | Lesen/Schreiben | 19-7 |
| CVP - Regelvariable (%) | PD10:0.CVP | Wort (INT) | 0 bis 100 | Steuerung | Lesen/Schreiben | 19-7 |
| SPV - Skalierte Prozessvariable | PD10:0.SPV | Wort (INT) | 0 bis 16383 | Status | Nur Lesen | 19-7 |

### Regelvariable (CV)

| Ausgangs-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| CV - Regel-variable | anwender-definiert | Wort (INT) | 0 bis 16383 | Steuerung | Lesen/Schreiben |

Die CV (Regelvariable) wird vom Anwender definiert. Siehe nachfolgenden Kontaktplanstrompfad.




### Regelvariable, % (CVP)

| Ausgangs-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| CVP - Regelvariable (%) | PD10:0.CVP | Wort (INT) | 0 bis 100 | Steuerung | Lesen/Schreiben |

CVP (Regelvariable, %) zeigt die Regelvariable als Prozentsatz an. Der Wert liegt in einem Bereich von 0 bis 100 %. Wenn das PD10:0/AM-Bit ausgeschaltet ist (Automatikmodus), folgt dieser Wert dem Regel-variablenausgang. Jegliche durch die Programmiersoftware geschrie- benen Werte werden überschrieben. Wenn das PD10:0/AM-Bit eingeschaltet ist (Manueller Modus), kann dieser Wert durch die Programmiersoftware eingegeben werden, und der Regelvariablen- ausgang folgt dem Regelvariablen-Prozentsatz.

### Skalierte Prozessvariable (SPV)

| Eingangs-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| SPV - Skalierte Prozessvariable | PD10:0.SPV | Wort (INT) | 0 bis 16383 | Status | Nur Lesen |

Die SPV (Skalierte Prozessvariable) ist die Analogausgangs-Variable. Bei aktivierter Skalierung bilden der Minimum-Sollwert (MINS) und der Maximum-Sollwert (MAXS) die Grenzen des Bereichs für diese Größe.

Ist der SPV-Wert so konfiguriert, dass er in physikalischen Einheiten gelesen werden kann, entspricht dieser Parameter dem Wert der Prozessvariablen in physikalischen Einheiten. Weitere Informationen zur Skalierung finden Sie im Abschnitt „Analog-E/A-Skalierung“ auf Seite 19-18.

## Abstimmparameter

Die nachfolgende Tabelle zeigt die Adressen, Datenformate und Anwenderprogrammzugriffe der Abstimmparameter. Beschreibungen der einzelnen Parameter finden Sie auf den in der letzten Spalte angegebenen Seiten.

| Abstimmparameter | Adresse | Datenformat | Bereich | Typ | Anwender-programm-zugriff | Weitere Informationen |
|---|---|---|---|---|---|---|
| KC - Verstärkung - $K_c$ | PD10:0.KC | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/ Schreiben | 19-9 |
| TI - Integralzeit - $T_i$ | PD10:0.Ti | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/ Schreiben | 19-9 |
| TD - Differenzialzeit - $T_d$ | PD 10:0.TD | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/ Schreiben | 19-9 |
| TM - Zeitmodus | PD10:0.TM | Binärwert | 0 oder 1 | Steuerung | Lesen/ Schreiben | 19-10 |
| LUT - Regelkreisaktualisierung | PD10:0.LUT | Wort (INT) | 1 bis 1024 | Steuerung | Lesen/ Schreiben | 19-10 |
| ZCD - Nulldurchgang-Totzone | PD10:0.ZCD | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/ Schreiben | 19-11 |
| FF - Störgrößenaufschaltung | PD10:0.FF | Wort (INT) | -16383 bis +16383 | Steuerung | Lesen/ Schreiben | 19-11 |
| SE - Skalierte Abweichung | PD10:0.SE | Wort (INT) | -32768 bis +32767 | Status | Nur Lesen | 19-11 |
| AM - Auto/Hand | PD10:0/AM | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/ Schreiben | 19-12 |
| CM - Regelmodus | PD10:0/CM | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/ Schreiben | 19-12 |
| DB - Totzonenfehler | PD10:0/DB | Binärwert (Bit) | 0 oder 1 | Status | Lesen/ Schreiben | 19-12 |
| RG - PLC 5-Verstärkungsbereich | PD10:0/RG | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/ Schreiben | 19-13 |
| SC - Sollwertskalierung | PD10:0/SC | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/ Schreiben | 19-13 |
| TF - Regelkreisaktualisierung zu schnell | PD10:0/TF | Binärwert (Bit) | 0 oder 1 | Status | Lesen/ Schreiben | 19-14 |
| DA - Differenzialhandlungs-Bit | PD10:0/DA | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/ Schreiben | 19-14 |
| UL - Oberer Ausgangsalarm | PD10:0/UL | Binärwert (Bit) | 0 oder 1 | Status | Lesen/ Schreiben | 19-14 |
| LL - Unterer Ausgangsalarm | PD10:0/LL | Binärwert (Bit) | 0 oder 1 | Status | Lesen/ Schreiben | 19-14 |
| SP - Sollwert nicht im Bereich | PD10:0/SP | Binärwert (Bit) | 0 oder 1 | Status | Lesen/ Schreiben | 19-15 |
| PV - Prozessvariable nicht im Bereich | PD10:0/PV | Binärwert (Bit) | 0 oder 1 | Status | Lesen/ Schreiben | 19-15 |
| DN - PID fertig | PD10:0/DN | Binärwert (Bit) | 0 oder 1 | Status | Nur Lesen | 19-15 |
| EN - PID aktiviert | PD10:0/EN | Binärwert (Bit) | 0 oder 1 | Status | Nur Lesen | 19-15 |
| IS - Integralsumme | PD10:0.IS | Doppelwort (32-Bit INT) | -2 147 483 648 bis 2 147 483 647 | Status | Lesen/ Schreiben | 19-15 |
| AD - Geänderter Differenzialanteil | PD10:0.AD | Doppelwort (32-Bit INT) | -2 147 483 648 bis 2 147 483 647 | Status | Nur Lesen | 19-16 |

## Verstärkung ($K_c$)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programm-zugriff |
|---|---|---|---|---|---|
| KC - Verstärkung - $K_c$ | PD10:0.KC | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/ Schreiben |

Der Verstärkungsfaktor $K_c$ (Wort 3) ist die Proportionalverstärkung, die in einem Bereich von 0 bis 3276,7 (bei RG = 0) oder von 0 bis 327,67 (bei RG = 1) liegt. Setzen Sie diesen Faktor auf die Hälfte des Werts, der für eine Oszillation des Ausgangs erforderlich ist, wenn die Integralzeit und die Differenzialzeit (siehe unten) auf null gesetzt sind.

**HINWEIS** Die Verstärkung wird durch das RG-Bit (Integralzeit- und Verstärkungsbereich) beeinflusst. Weitere Informationen hierzu finden Sie unter „PLC 5-Verstärkungsbereich (RG)“ auf Seite 19-13.

## Integralzeit ($T_i$)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programm-zugriff |
|---|---|---|---|---|---|
| TI - Integralzeit - $T_i$ | PD10:0.Ti | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/ Schreiben |

Die Integralzeit $T_i$ (Wort 4) ist die Integralverstärkung, die in einem Bereich von 0 bis 3276,7 (bei RG = 0) oder von 0 bis 327,67 (bei RG = 1) Minuten pro Wiederholung liegt. Setzen Sie die Integralzeit auf die in der obigen Verstärkungskalibrierung gemessene Zeit. Der Wert 1 fügt der PID-Gleichung die maximale Integralzeit hinzu.

**HINWEIS** Die Integralzeit wird durch das RG-Bit (Integralzeit- und Verstärkungsbereich) beeinflusst. Weitere Informationen hierzu finden Sie unter „PLC 5-Verstärkungsbereich (RG)“ auf Seite 19-13.

## Differenzialzeit ($T_d$)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| TD - Differenzialzeit - $T_d$ | PD 10:0.TD | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/Schreiben |

Die Differenzialzeit $T_d$ (Wort 5) ist der Differenzialterm. Der einstell- bare Bereich beträgt 0 bis 327,67 Minuten. Setzen Sie diesen Wert auf 1/8 der obigen Integralzeit $T_i$.

**HINWEIS** Dieses Wort wird nicht durch das RG-Bit (Integralzeit- und Verstärkungsbereich) beeinflusst. Weitere Infor- mationen hierzu finden Sie unter „PLC 5-Verstärkungsbereich (RG)“ auf Seite 19-13.

## Zeitmodus (TM)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| TM - Zeitmodus | PD10:0.TM | Binärwert | 0 oder 1 | Steuerung | Lesen/Schreiben |

Das Zeitmodus-Bit zeigt an, wann sich der PID-Befehl im Zeitregelungsmodus (1) oder im STI-Modus (0) befindet. Dieses Bit kann durch Befehle im Kontaktplan gesetzt bzw. zurückgesetzt werden.

Im Zeitsteuerungsmodus wird die Regelvariable mit der durch den Regelkreisaktualisierungsparameter (PD10:0.LUT) festgelegten Geschwindigkeit durch PID aktualisiert.

Im STI-Modus wird die Regelvariable bei jeder Abfrage des PID-Befehls im Regelprogramm durch PID aktualisiert. Im STI-Modus sollte der Befehl in ein STI-Interrupt-Unterprogramm integriert werden. Die STI-Routine sollte ein Zeitintervall aufweisen, das mit der Einstellung des PID-Regelkreisaktualisierungsparameters (PD10:0.LUT) identisch ist. Geben Sie das STI-Intervall in das Wort STI:0.SPM ein. Wurde der Regelkreisaktualisierungszeit beispielsweise der Wert 10 (= 100 ms) zugewiesen, muss das STI-Zeitintervall ebenfalls 100 (= 100 ms) betragen.

**HINWEIS** Im Zeitsteuerungsmodus sollte die Abfragezeit des Prozessors mindestens zehnmal kürzer sein als die Regelkreisaktualisierungszeit, um Ungenauigkeiten oder Störungen aufgrund der Zeitsteuerung zu vermeiden.

## Regelkreisaktualisierung (LUT)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| LUT - Regelkreis-aktualisierung | PD10:0.LUT | Wort (INT) | 1 bis 1024 | Steuerung | Lesen/Schreiben |

Die Regelkreisaktualisierungszeit (Wort 13) ist das Zeitintervall zwischen den PID-Rechenoperationen. Der Eintrag erfolgt in Intervallen von 0,01 Sekunden. Geben Sie eine Regelkreisaktu- alisierungszeit ein, die fünf- bis zehnmal kürzer ist als die Eigenperiode der Last. Die Eigenperiode der Last können Sie bestimmen, indem Sie den Integralzeit- und den Differenzialzeitpa- rameter auf null setzen und dann die Verstärkung erhöhen, bis die Ausgangsoszillation einsetzt. Im STI-Modus muss dieser Wert dem STI-Zeitintervall in STI:0.SPM entsprechen. Die Werte müssen in einem Bereich von 0,01 bis 10,24 Sekunden liegen.

## Nulldurchgang-Totzone (ZCD)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| ZCD - Nulldurchgang-Totzone | PD10:0.ZCD | Wort (INT) | 0 bis 32767 | Steuerung | Lesen/Schreiben |

Die Totzone umfasst den Bereich oberhalb und unterhalb des Soll- werts; die Größe dieses Bereiches wird durch den eingegebenen Wert bestimmt. Die Totzone wird beim Nulldurchgang der Prozessvariablen und des Sollwerts eingegeben. Dies bedeutet, dass die Totzone erst wirksam ist, wenn die Prozessvariable innerhalb dieses Bereiches liegt *und* den Sollwert durchlaufen hat.

Gültige Werte liegen in einem Bereich von 0 bis zum skalierten Maximum oder von 0 bis 16.383, wenn keine Skalierung vorge- nommen wurde.

## Störgrößenaufschaltung (FF)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| FF - Störgrößen-aufschaltung | PD10:0.FF | Wort (INT) | -16383 bis +16383 | Steuerung | Lesen/Schreiben |

Mit der Störgrößenaufschaltung werden Störungen ausgeglichen, die den Regelvariablenausgang beeinträchtigen können.

## Skalierte Abweichung (SE)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| SE - Skalierte Abweichung | PD10:0.SE | Wort (INT) | -32768 bis +32767 | Status | Nur Lesen |

Der Skalierte Abweichung gibt die Differenz zwischen der Prozess- variablen und dem Sollwert an. Das Format der Differenz (E = SP-PV oder E = PV-SP) wird durch das CM-Bit (Regelmodus) bestimmt. Siehe „Regelverhalten (CM)“ auf Seite 19-12.

## Automatisch/manuell (AM)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| AM - Auto/Hand | PD10:0/AM | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/Schreiben |

Das AM-Bit kann durch Befehle im Kontaktplan gesetzt bzw. zurückgesetzt werden. Bei ausgeschaltetem Bit (0) ist der Automa- tikbetrieb aktiviert. Bei eingeschaltetem Bit (1) ist der Handbetrieb aktiviert. Im Automatikbetrieb regelt der Befehl die Regelvariable (CV). Im Handbetrieb wird die Regelvariable durch das Anwender-/Regelprogramm geregelt. Setzen Sie dieses Bit während der Abstimmung auf manuell.

**HINWEIS** Die Ausgangsbegrenzung wird ebenfalls im Hand- betrieb angewendet.

## Regelverhalten (CM)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| CM - Regelmodus | PD10:0/CM | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/Schreiben |

Mit dem Regelverhalten (oder öffnend/schließend) können Sie umschalten zwischen den Werten E=SP-PV und E=PV-SP.

Bei der öffnenden Steuerung (E=PV-SP) wird die Regelvariable erhöht, wenn die Prozessvariable größer ist als der Sollwert.

Bei der schließenden Steuerung (E=SP-PV) wird die Regelvariable reduziert, wenn die Prozessvariable größer ist als der Sollwert.

## Totzonenfehler (DB)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| DB - Totzonenfehler | PD10:0/DB | Binärwert (Bit) | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt (1), wenn sich die Prozessvariable innerhalb des Nulldurchgang-Totzonenbereichs befindet.

## PLC 5-Verstärkungsbereich (RG)

| Abstimm-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| RG - PLC 5-Verstärkungs-bereich | PD10:0/RG | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/Schreiben |

Wenn dieses Bit gesetzt (1) ist, werden der Wert 'Minuten rücksetzen/ wiederholen' und der Verstärkungsmultiplikator (KC) durch das Rück- setz- und Verstärkungsbereichserweiterungs-Bit (TI- und RG-Bit) durch 10 dividiert. Damit ergeben sich ein Integralmultiplikator von 0,01 und ein Verstärkungsmultiplikator von 0,01.

Wenn dieses Bit zurückgesetzt (0) ist, können der Wert 'Minuten rücksetzen/ wiederholen' und der Verstärkungsmultiplikator mit einem Integralmultiplikator 0,1 und Verstärkungsmultiplikator 0,1 bewertet werden.

*Beispiel, RG-Bit gesetzt*: Die Integralzeit (TI) 1 zeigt an, dass auf den PID-Integralalgorithmus der Integralwert 0,01 Minuten/Wiederholung (0,6 Sekunden/Wiederholung) angewendet wird. Der Verstärkungs- wert (KC) 1 zeigt an, dass der Fehler mit 0,01 multipliziert und auf den PID-Algorithmus angewendet wird.

*Beispiel, RG-Bit gelöscht*: Die Integralzeit (TI) 1 zeigt an, dass auf den PID-Integralalgorithmus der Integralwert 0,1 Minuten/Wiederholung (6,0 Sekunden/Wiederholung) angewendet wird. Der Verstärkungs- wert (KC) 1 zeigt an, dass der Fehler mit 0,1 multipliziert und auf den PID-Algorithmus angewendet wird.

**HINWEIS** Der Differenzialmultiplikator (TD) wird durch diese Auswahl nicht beeinflusst.

## Sollwertskalierung (SC)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| SC - Sollwertskalierung | PD10:0/SC | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/Schreiben |

Das SC-Bit wird gelöscht, wenn Sollwertskalierwerte angegeben werden.

## Regelkreisaktualisierung zu schnell (TF)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| TF - Regelkreis-aktualisierung zu schnell | PD10:0/TF | Binärwert (Bit) | 0 oder 1 | Status | Lesen/Schreiben |

Das TF-Bit wird durch den PID-Algorithmus gesetzt, wenn die angegebene Regelkreisaktualisierungszeit aufgrund von Abfragezeit- beschränkungen nicht durch die Steuerung umgesetzt werden kann.

Wenn dieses Bit gesetzt ist, kann die Störung durch die Aktualisierung des PID-Regelkreises in längeren Zeitabständen oder durch die Ver- schiebung des PID-Befehls in ein STI-Interrupt-Unterprogramm behoben werden. Ist dieses Bit während der Ausführung des Befehls gesetzt, treten Fehler bei der Integralzeit und der Differenzial- verstärkung auf.

## Differenzialverhalten-Bit (DA)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| DA - Differenzial-handlungs-Bit | PD10:0/DA | Binärwert (Bit) | 0 oder 1 | Steuerung | Lesen/Schreiben |

Wenn dieses Bit gesetzt (1) ist, wird die Differenzialberechnung anhand der Abweichung anstelle der Prozessvariablen bewertet. Wenn dieses Bit zurückgesetzt (0) ist, kann die Differenzialberechnung anhand der Prozessvariablen erfolgen.

## Oberer Ausgangsalarm (UL)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| UL - Oberer Ausgangsalarm | PD10:0/UL | Binärwert (Bit) | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt, wenn der berechnete Regelvariablenausgang die Obergrenze für die Regelvariable überschreitet.

## Unterer Ausgangsalarm (LL)

| Abstimmparameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| LL - Unterer Ausgangsalarm | PD10:0/LL | Binärwert (Bit) | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt, wenn der berechnete Regelvariablenausgang die Untergrenze für die Regelvariable unterschreitet.

## Sollwert nicht im Bereich (SP)

| Abstimmparameter | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| SP - Sollwert nicht im Bereich | PD10:0/SP | Binärwert (Bit) | 0 oder 1 | Status | Lesen/ Schreiben |

Dieses Bit wird gesetzt (1), wenn der Sollwert:

- den skalierten Höchstwert überschreitet oder
- den skalierten Mindestwert unterschreitet.

## Prozessvariable nicht im Bereich (PV)

| Abstimmparameter | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| PV - Prozessvariable nicht im Bereich | PD10:0/PV | Binärwert (Bit) | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt (1), wenn die nicht skalierte Prozessvariable

- den Wert von 16383 überschreitet oder
- kleiner 0 ist.

## Fertig (DN)

| Abstimmparameter | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| DN - PID fertig | PD10:0/DN | Binärwert (Bit) | 0 oder 1 | Status | Nur Lesen |

Dieses Bit wird bei der Berechnung des PID-Algorithmus bei einer Abfrage gesetzt (1). Es wird zurückgesetzt (0), wenn der Befehl abgefragt und der PID-Algorithmus nicht berechnet wurde (gilt nur für den zeitgesteuerten Modus).

## Aktiviert (EN)

| Abstimmparameter | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| EN - PID aktiviert | PD10:0/EN | Binärwert (Bit) | 0 oder 1 | Status | Nur Lesen |

Dieses Bit wird gesetzt (1), wenn der PID-Befehl aktiviert wurde. Es folgt dem Strompfadstatus.

## Integralsumme (IS)

| Abstimmparameter | Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|---|
| IS - Integralsumme | PD10:0.IS | Doppelwort (32-Bit INT) | -2 147 483 648 bis 2 147 483 647 | Status | Lesen/Schreiben |

Dies ist das Ergebnis der Integralberechnung $\frac{K_C}{T_I}\int E(dt)$.

## Geänderter Differenzialanteil (AD)

| Abstimm-parameter | Adresse | Daten-format | Bereich | Typ | Anwender-programmzugriff |
|---|---|---|---|---|---|
| AD - Geänderter Differenzialanteil | PD10:0.AD | Doppelwort (32-Bit INT) | -2 147 483 648 bis 2 147 483 647 | Status | Nur Lesen |

Dieses Doppelwort wird intern zur Verfolgung der Änderung der Prozessvariablen während der Regelkreisaktualisierungszeit verwendet.

## Laufzeitfehler

Im Falle eines Laufzeitfehlers des PID-Befehls wird im Statusfile der Fehlercode 0036 angezeigt. Dieser Code kennzeichnet die folgenden PID-Fehlerzustände, die jeweils einem bestimmten EinzelbyteCodewert zugewiesen wurden. Dieser erscheint im MSByte des zweiten Worts im Regelblock.

| Fehlercode | Beschreibung des Fehlerzustands bzw. der Fehlerzustände | | Abhilfemaßnahme | |
|---|---|---|---|---|
| **11H** | 1. Regelkreisaktualisierungszeit<br>$D_t > 1024$ | | Regelkreisaktualisierungszeit ändern $0 < D_t \leq 1024$ | |
| | 2. Regelkreisaktualisierungszeit<br>$D_t = 0$ | | | |
| **12H** | Proportionalverstärkung<br>$K_c < 0$ | | Proportionalverstärkung ändern $K_c$ auf $0 < K_c$ | |
| **13H** | Integralverstärkung (Integralanteil)<br>$T_i < 0$ | | Integralverstärkung (Integralanteil) ändern $T_i$ auf $0 \leq T_i$ | |
| **14H** | Differenzialverstärkung (Differenzialanteil)<br>$T_d < 0$ | | Differenzialverstärkung (Differenzialanteil) ändern $T_d$ auf $0 \leq T_d$ | |
| **15H** | FF (Feed Forward Bias) liegt außerhalb des Bereichs. | | Ändern Sie FF so, dass der Wert zwischen -16383 und +16383 liegt. | |
| **23H** | Skalierter min. Sollwert<br>MinS > Skalierter max. Sollwert MaxS | | Skalierten min. Sollwert ändern MinS auf<br>-32768 ≤ MinS ≤ MaxS ≤ +32767 | |
| **31H** | Wenn Sie die Sollwertskalierung verwenden und MinS > Sollwert SP > MaxS oder<br><br>Wenn Sie die Sollwertskalierung nicht verwenden und 0 > Sollwert SP > 16383,<br><br>tritt während der ersten Ausführung des PID-Regelkreises dieser Fehler auf und Bit 11 in Wort 0 des Regelblocks wird gesetzt. Wird dagegen bei der darauf folgenden Ausführung des PID-Regel- kreises ein ungültiger Regelkreis-Sollwert spezifiziert, wird die Ausführung des PID-Regelkreises mit dem alten Sollwert fortgesetzt und Bit 11 in Wort 0 des Regelblocks gesetzt. | | Wenn Sie die Sollwertskalierung verwenden, ändern Sie den Sollwert SP wie folgt: MinS ≤ SP ≤ MaxS oder<br><br>Wenn Sie die Sollwertskalierung nicht verwenden, ändern Sie den Sollwert SP wie folgt: 0 ≤ SP ≤ 16383. | |
| **41H** | **Skalierung ausgewählt** | **Skalierung nicht ausgewählt** | **Skalierung ausgewählt** | **Skalierung nicht ausgewählt** |
| | 1. Totzone < 0 oder | 1. Totzone < 0 oder | Ändern Sie die Totzone wie folgt:<br>0 ≤ Totzone ≤<br>(MaxS – MinS) ≤ 16383 | Ändern Sie die Totzone wie folgt:<br>0 ≤ Totzone ≤ 16383 |
| | 2. Totzone > (MaxS – MinS) | 3. Totzone > 16383 | | |
| **51H** | 1. Obere Ausgangsgrenze < 0 oder<br>2. Obere Ausgangsgrenze > 100 | | Ändern Sie die obere Ausgangsgrenze wie folgt:<br>0 ≤ obere Ausgangsgrenze ≤ 100 | |
| **52H** | 1. Untere Ausgangsgrenze < 0 oder<br>2. Untere Ausgangsgrenze > 100 | | Ändern Sie die untere Ausgangsgrenze wie folgt:<br>0 ≤ untere Ausgangsgrenze ≤ obere Ausgangsgrenze ≤ 100 | |
| **53H** | Untere Ausgangsgrenze > obere Ausgangsgrenze | | Ändern Sie die untere Ausgangsgrenze wie folgt:<br>0 ≤ untere Ausgangsgrenze ≤ obere Ausgangsgrenze ≤ 100 | |

## Analog-E/A-Skalierung

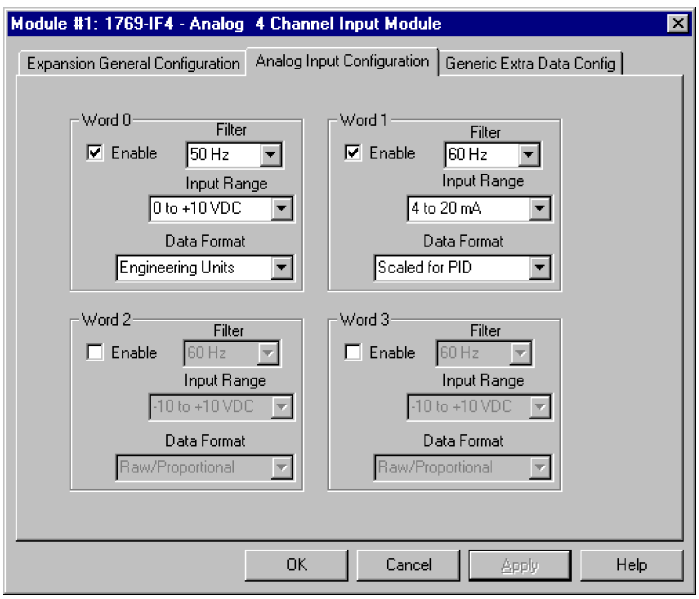
Bei der Konfiguration eines Analogausgangs für die Verwendung eines PID-Befehls müssen die Analogdaten skaliert und damit an die PID-Befehlsparameter angeglichen werden. Bei MicroLogix 1200- und MicroLogix 1500-Steuerungen ist die Prozessvariable (PV) des PID-Befehls auf einen Datenbereich von 0 bis 16383 ausgelegt. Die Compact-E/A-Analogmodule 1769 (1769-IF4 und 1769-OF2) bieten die Möglichkeit der Skalierung von Analogdaten direkt auf dem Modul. Bei der Skalierung von Daten wird der Bereich des Analogeingangs an den Eingangsbereich des PID-Befehls angepasst. Die Möglichkeit, die Skalierung in den E/A-Modulen durchführen zu können, reduziert den Programmieraufwand im System und erleichtert die PID-Konfiguration erheblich.

Das Beispiel zeigt ein Modul 1769-IF4. Dieses Modul verfügt über vier separat konfigurierbare Eingänge. In diesem Beispiel ist der Analog- eingang 0 für 0 bis 10 V konfiguriert und in physikalischen Einheiten skaliert. Das Wort 0 wird in dem PID-Befehl nicht verwendet. Eingang 1 (Wort 1) ist für einen Betrieb mit 4 bis 20 mA und die Skalierung für einen PID-Befehl konfiguriert. Damit werden die Analogdaten für den PID-Befehl konfiguriert.

| Eingangssignal von Feldgerät | Analogregister-Skalierdaten |
|---|---|
| > 20,0 mA | 16384 bis 17406 |
| 20,0 mA | 16,383 |
| 4,0 mA | 0 |
| < 4,0 mA | -819 bis -1 |

Der Analog-Konfigurationsbildschirm wird in RSLogix 500 aufgerufen. Klicken Sie im Ordner „Controller“ (Steuerung) auf den Eintrag „I/O Configuration“ (E/A-Konfiguration), und doppelklicken Sie dann auf das betreffende E/A-Modul.

Die Konfiguration für Analogausgänge ist praktisch mit der beschrie- benen Vorgehensweise identisch. Die PID-Regelvariable wird einfach an die Analogausgangsadresse adressiert; dann wird der Analogaus- gang für die PID-Skalierung konfiguriert.



## Anwendungshinweise

In den folgenden Abschnitten werden diese Themen behandelt:

- Eingangs-/Ausgangsbereiche
- Skalierung auf technische Einheiten
- Totzonen-Nulldurchgang
- Ausgangsalarme
- Ausgangsbegrenzung mit Anti-Reset-Windup
- Manueller Modus
- Störgrößenaufschaltung

**ACHTUNG**

Ändern Sie den Zustand eines beliebigen PID-Steuerungsblockwerts erst, wenn Sie die Funktion und ihre Auswirkung auf Ihren Prozess vollständig verstanden haben. Eine unerwartete Operation kann zu einer Beschädigung der Ausrüstung oder zu Verletzungen führen.

### Eingangs/Ausgangsbereiche

Das Eingangsmodul zur Messung der Prozessvariablen (PV) muss über einen vollständigen Binärwertbereich von 0 bis 16383 verfügen. Wenn dieser Wert kleiner als 0 ist (Bit 15 gesetzt), wird der Wert 0 für die Prozessvariable verwendet und das Bit „Prozessvariable nicht im Bereich“ gesetzt (Bit 12 in Wort 0 des Regelblocks). Ist die Prozess- variable größer als 16383 (Bit 14 gesetzt), wird der Wert 16383 für die Prozessvariable verwendet und das Bit „Prozessvariable nicht im Bereich“ gesetzt.

Die durch den PID-Befehl berechnete Regelvariable weist denselben Bereich von 0 bis 16383 auf. Der Steuerungsausgang (Wort 16 des Regelblocks) kann Werte von 0 bis 100 % annehmen. Für die durch den Befehl berechneten Ausgangswerte können eine Unter- und eine Obergrenze festgelegt werden (dabei entspricht eine Obergrenze von 100 % einer Regelvariablengrenze von 16383).

## Skalierung auf technische Einheiten

Mit Hilfe der Skalierung können Sie den Sollwert und den Wert für den Totzonen-Nulldurchgang in physikalischen Einheiten eingeben und die Prozessvariable und die Abweichungswerte in denselben physikalischen Einheiten anzeigen. Beachten Sie, dass die Prozess- variable PV in einem Bereich zwischen 0 und 16383 liegen muss. Die Prozessvariable wird in physikalischen Einheiten angezeigt.

Gehen Sie bei der Wahl der Skalierung wie folgt vor:

1. Geben Sie im PID-Regelblock den Höchst- und Mindestskalierwert (MaxS und MinS) ein. Der MinS-Wert entspricht einem Analogwert von Null für den niedrigsten Wert der Prozessvariablen. MaxS entspricht einem Analogwert von 16383 für den höchsten Wert. Diese Werte kennzeichnen die Prozessgrenzwerte. Die Sollwert- skalierung wird durch Eingabe eines Wertes ungleich null für einen oder beide Parameter ausgewählt. Wenn Sie für beide Parameter den gleichen Wert eingeben, wird die Sollwertskalierung deaktiviert.

   Beispiel: Geben Sie bei der Messung eines vollständigen Temperaturbereichs von –73 °C (PV = 0) bis +1156 °C (PV = 16383) den Wert –73 als MinS und 1156 als MaxS ein. Beachten Sie, dass PID-Befehlseingänge in einem Bereich von 0 bis 16383 liegen müssen. Signale könnten wie folgt umgesetzt werden:

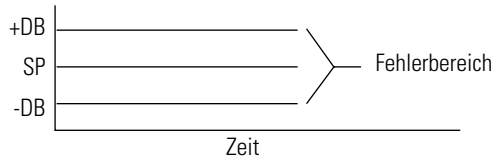
| **Beispielwerte** | |
|---|---|
| Prozessgrenzwerte | –73 bis+1156 °C |
| Signalwandler-Ausgang (sofern verwendet) | +4 bis +20 mA |
| Ausgang des AnalogEingangsmoduls | 0 bis 16383 |
| PID-Befehl, MinS bis MaxS | –73 bis +1156 °C |

2. Geben Sie den Sollwert (Wort 2) und die Totzone(Wort 9) in den gleichen skalierten physikalischen Einheiten ein. Das Auslesen der skalierten Prozessvariablen und der skalierten Abweichung erfolgt ebenfalls in diesen Einheiten. Der Steuerungsausgang (Wort 16) wird als Prozentualwert des Bereiches zwischen 0 und 16383 angezeigt. Der an den Ausgang übertragene Wert liegt immer zwischen 0 und 16383.

Bei Verwendung der Skalierung werden Sollwert, Totzone, Prozess- variablen und Abweichung durch den Befehl skaliert. Wenn Sie die Skalierung ändern, sollten Sie die Auswirkungen auf alle diese Variablen berücksichtigen.

## Nulldurchgang-Totzone DB

Die einstellbare Totzone ermöglicht die Festlegung eines Abweichungsbereichs oberhalb und unterhalb des Sollwerts, wobei der Ausgang unverändert bleibt, solange die Abweichung innerhalb dieses Bereichs liegt. Auf diese Weise können Sie überprüfen, wie weit die Prozessvariable vom Sollwert abweicht, ohne den Ausgang zu verändern.




Der Nulldurchgang ist eine Totzonen-Überwachung, aufgrund derer der Befehl für Berechnungen die Abweichung vom Eintreten der Prozessvariablen in die Totzone bis zum Erreichen des Sollwertes verwendet. Durchläuft die Abweichung die Sollwertgrenze (Null-durchgang der Abweichung und Vorzeichenänderung) und bleibt er innerhalb der Totzone, weist der Befehl bei der Ausführung von Berechnungen der Abweichung den Wert null zu.

Legen Sie die Totzone durch die Eingabe eines Wertes in das Totzonen-Speicherwort (Wort 9) des Regelblocks fest. Die Totzone umfasst den Bereich oberhalb und unterhalb des Sollwerts. Die Größe dieses Bereiches wird durch den eingegebenen Wert bestimmt. Der Wert 0 deaktiviert diese Funktion. Bei einer Skalierung entsprechen die skalierten Einheiten der Totzone denen des Sollwertes.

## Ausgangsalarme

Sie können für die Regelvariable an einem gewählten Wert oberhalb und/oder unterhalb eines bestimmten Ausgangsprozentualwertes einen Ausgangsalarm spezifizieren. Wenn festgestellt wird, dass die Regelvariable einen gesetzten Wert über- oder unterschreitet, wird in dem PID-Befehl ein Alarm-Bit (LL-Bit für unteren Ausgangsalarm, UL-Bit für oberen Ausgangsalarm) gesetzt. Alarm-Bits werden durch den Befehl zurückgesetzt, wenn die Regelvariable wieder innerhalb der Grenzwerte liegt. Sofern die Ausgangsbegrenzung nicht ausgewählt wurde, wird das Über- oder Unterschreiten der Alarmwerte durch die Regelvariable nicht verhindert.

Die Wahl des oberen und unteren Ausgangsalarms erfolgt durch die Eingabe einer Obergrenze (CVH) und einer Untergrenze (CVL). Die Grenzwerte werden als Prozentualwert des Ausgangs spezifiziert. Sollen keine Alarme festgelegt werden, geben Sie für die Untergrenze 0 und die Obergrenze 100 % ein, und ignorieren Sie die Alarm-Bits.

## Ausgangsbegrenzung mit Anti-Reset-Windup

Sie können für die Regelvariable einen Ausgangsbegrenzung (in Prozent des Ausgangs) festlegen. Wenn der Befehl feststellt, dass die Regelvariable einen Grenzwert über- oder unterschreitet, wird ein Alarm-Bit (LL-Bit für Untergrenze, UL-Bit für Obergrenze) gesetzt und das Überschreiten des Grenzwerts durch die Regelvariable verhindert. Wenn Sie keine Grenzwerte festlegen, wird die Regelvariable durch 0 und 100 % begrenzt.

Die Unter- und Obergrenze für den Ausgang können Sie durch Setzen des OL-Bits (Ausgangsbegrenzung) und Eingabe einer Obergrenze (CVH) und einer Untergrenze (CVL) auswählen. Grenzwerte werden in Prozent (0 bis 100 %) der Regelvariablen angegeben.

Der Unterschied zwischen dem Wählen von Ausgangsalarmen und dem Wählen von Ausgangsbegrenzungen besteht darin, dass Sie zur Aktivierung von Grenzwerten die Ausgangsbegrenzung auswählen müssen. Grenz- und Alarmwerte werden in den gleichen Worten gespeichert. Bei Eingabe dieser Werte werden zwar Alarme, nicht jedoch die Begrenzung aktiviert. Wenn Sie Grenzwerte angeben und darüber hinaus das Grenzwert-Freigabe-Bit setzen, werden sowohl die Begrenzung als auch der Alarm aktiviert.

Die Anti-Reset-Windup-Funktion verhindert ein übermäßiges Ansteigen des Integralterms, wenn die Regelvariable einen Grenzwert erreicht. Wenn die Summe des PID und der Kompensation in der Regelvariablen den Grenzwert erreicht, wird die Berechnung der Integralsumme unterbrochen, bis die Regelvariable wieder innerhalb des gültigen Bereichs liegt. Die Integralsumme ist in dem Element IS gespeichert.

## Manueller Modus

Im manuellen Modus wird der Wert der Regelvariablen nicht durch den PID-Algorithmus berechnet. Statt dessen wird der Wert als Eingang zur Anpassung der Integralsumme (IS) verwendet, damit bei der Aktivierung des Automatikmodus ein stoßfreier Übergang sichergestellt ist.

Im manuellen Modus können Sie einen neuen CV-Wert von 0 bis 100 % eingeben. Dieser Wert wird in eine Zahl zwischen 0 und 16383 umgewandelt und in die Adresse der Regelvariablen gestellt. Wenn Sie die manuelle Ausgangsgröße über den Kontaktplan festlegen, muss der Kontaktplan so strukturiert sein, dass er Daten im manuellen Modus in die CV-Adresse schreibt. Beachten Sie, dass der neue Wert der Regelvariablen im Bereich von 0 bis 16383, nicht von 0 bis 100 liegt. Die Eingabe des Prozentwertes der Regelvariablen (CVP) in den Kontaktplan wird im manuellen Modus ignoriert.

## PID-Strompfadstatus

Wenn der PID-Strompfad unwahr ist, wird die Integralsumme (IS) gelöscht, und die Regelvariable behält ihren letzten Status bei.

### Störgrößenaufschaltung

Anwendungen mit Transportverzögerungen erfordern zur Vorbeugung einer Störung ggf. eine Kompensation des CV-Ausgangs. Diese Kompensation kann durch Eingabe eines Werts in dem FF-Element (Störgrößen oder Grundlast) erreicht werden (Siehe Seite 19-11.) Der angegebene Wert wird zu dem Ausgang addiert, wodurch eine Störgrößenaufschaltung stattfindet. Die Aufschaltung kann durch Eingabe eines Werts von -16383 bis +16383 in Wort 6 über das Programmiergerät oder den Kontaktplan aktiviert werden.

## Anwendungsbeispiele

### PID-Regelkreis einstellen

Das Einstellen eines PID-Regelkreises erfordert Kenntnisse im Bereich der Prozessregelung. Wenn Sie nicht über die entsprechende Erfah- rung verfügen, können Sie durch die Teilnahme an einer Schulung die Theorie sowie die in Ihrem Unternehmen eingesetzten Prozessregelungsverfahren erlernen.

Es gibt zahlreiche Verfahren zur Einstellung eines PID-Regelkreises. Das folgende allgemeine PID-Einstellverfahren bezieht sich ausschließlich auf die Handhabung von Laststörungen. Beim Einstellen empfiehlt es sich, die Änderungen im manuellen Modus vorzunehmen und anschließend in den Automatikmodus zurückzu- kehren. Im manuellen Modus ist die Ausgangsbegrenzung aktiviert.

**HINWEIS**

- Bei dieser Methode muss der PID-Befehl eine Anwendung regeln, bei der kein außergewöhnliches Verletzungs- oder Sachschadensrisiko besteht.
- Das PID-Einstellverfahren funktioniert unter Umständen nicht in allen Fällen. Die besten Ergebnisse erzielen Sie mit einem PID-Regelkreis- einstellpaket (z. B. RSTune, Rockwell Software-Bestellnummer 9323-1003D).

*Vorgehensweise*

1. Erstellen Sie den Kontaktplan. Der Analogeingang muss für den Bereich der Prozessvariablen (PV) und die Prozessvariable für den Analogausgang richtig skaliert sein.
2. Schließen Sie die Prozessregelgeräte an die Analogmodule an. Laden Sie das Programm in den Prozessor herunter. Belassen Sie den Prozessor im Programmier-Modus.

**ACHTUNG**

Stellen Sie sicher, dass zum Schutz von Personen und Anlagen alle möglichen Maschinenbewegungen berücksichtigt wurden. Es ist möglich, dass der Ausgang (CV) während der Einstellung zwischen 0 und 100 % schwingt.

**HINWEIS** Wie Sie die Skalierung Ihres kontinuierlichen Systems überprüfen und/oder die erste Regelkreisaktualisie- rungszeit Ihres Systems bestimmen können, entnehmen Sie bitte den Anweisungen auf Seite 19-26.

3. Geben Sie die folgenden Werte ein: den Anfangs-Sollwert SP, die Integralzeit $T_i$ 0, die Differenzialzeit $T_d$ 0, den Verstärkungsfaktor $K_c$ 1 und die Regelkreisaktualisierung 5.

   Wählen Sie im Kontaktplan den STI-Modus oder den Zeit-steuerungsmodus aus. Achten Sie bei der Wahl von STI darauf, dass die Regelkreisaktualisierungszeit dem STIZeitintervall entspricht.

   Geben Sie die gewünschten optionalen Einstellungen ein (Ausgangsbegrenzung, Ausgangsalarm, Smax - SminSkalierung, Störgrößenaufschaltung).
4. Stellen Sie den Zusammenhang zwischen dem zeitlichen Verlauf von CV, PV, Analog-Eingang bzw. Ausgang und dem Sollwert SP graphisch dar.
5. Wählen Sie für den PID-Befehl den manuellen Modus und für den Prozessor den Run-Modus aus.
6. Stellen Sie bei der Überwachung der PID-Anzeige den Prozess durch Korrektur des CO-Prozentualwertes manuell ein.
7. Wählen Sie für den PID-Befehl den Automatikmodus, wenn Sie den Prozess manuell unter Kontrolle haben.
8. Stellen Sie die Verstärkung ein. Beobachten Sie hierzu das Verhältnis zwischen dem Ausgang und dem Sollwert über einen längeren Zeitraum.

**9.** Wenn Sie bemerken, dass der Prozess regelmäßig ober- und unterhalb des Sollwertes schwingt, messen Sie die für einen Zyklus erforderliche Zeit. Sie erhalten damit die natürliche Schwingungsdauer des Prozesses.

Natürliche Zeitspanne ≅ 4x Totzeit

Zeichnen Sie den Wert für die Verstärkung auf. Kehren Sie in den manuellen Modus zurück (unterbrechen Sie gegebenenfalls den Prozess).

**10.** Stellen Sie die Regelkreisaktualisierungszeit (und gegebenenfalls das STI-Zeitintervall) auf einen gegenüber der natürlichen Schwingungsdauer fünf- bis zehnmal kleineren Wert ein.

Beispiel: Wenn die Zykluszeit 20 Sekunden beträgt und die Regelkreisaktualisierung zehnmal kürzer als die natürliche Schwingungsdauer sein soll, setzen Sie die Regelkreisaktua- lisierungszeit auf 200, was zu einem Zwei-Sekunden-Intervall führt.

**11.** Stellen Sie die Verstärkung $K_c$ auf die Hälfte des Wertes ein, der für die natürliche Schwingungsdauer des Prozesses erforderlich ist. Beträgt die in Schritt 9 ermittelte Verstärkung beispielsweise 80, stellen Sie die Verstärkung auf 40 ein.

**12.** Stellen Sie die Integralzeit $T_i$ ungefähr auf die natürliche Schwingungsdauer ein. Beträgt die natürliche Schwingungsdauer, wie in unserem Beispiel, 20 Sekunden, stellen Sie die Integralzeit auf 3 ein (0,3 Minuten pro Wiederholung entspricht etwa 20 Sekunden).

**13.** Stellen Sie jetzt die Differenzialzeit $T_d$ auf 1/8 der Integralzeit ein. In unserem Beispiel wird für die Differenzialzeit von 0,04 Minuten pro Wiederholung der Wert 4 eingegeben.

**14.** Aktivieren Sie den Automatikmodus für den Prozess. Im Falle eines idealen Prozesses ist die PID-Einstellung damit abge- schlossen.

**15.** Um an dieser Stelle weitere Einstellungen vorzunehmen, aktivie- ren Sie den manuellen PID-Modus, geben die entsprechenden Werte ein und schalten wieder auf Automatik um.

Durch das Umschalten zwischen manuellem und Automatikmodus wird sichergestellt, dass fast alle Verstärkungsfehler bei jeder Einstellung gelöscht werden. Somit können die Auswirkungen der Einstellungen sofort erkannt werden. Durch Umschalten des PID-Strompfads wird der PID-Befehl automatisch neu gestartet, so dass der gesamte Integralanteil gelöscht wird. Es empfiehlt sich, während des Einstellens den PID-Strompfad auf unwahr zu stellen, um die Auswirkungen von zuvor vorgenommenen Justierungen zu vermeiden.

## Bestätigung der Skalierung eines kontinuierlichen Systems

Um sicherzustellen, dass der jeweilige Prozess linear ist und die Geräte ordnungsgemäß angeschlossen und skaliert sind, müssen die folgenden Schritte durchgeführt werden:

1. Stellen Sie den manuellen PID-Modus ein, und geben Sie folgende Parameter ein:
   - Geben Sie ein: `0` bei MinS
   - Geben Sie ein: `100` bei MaxS
   - Geben Sie ein: `0` bei CO%
2. Schalten Sie in den dezentralen Run-Modus (REM Run) um, und vergewissern Sie sich, dass PV=0 ist.
3. Geben Sie ein: `20` bei CO%
4. Notieren Sie den Wert PV = _____
5. Geben Sie ein: `40` bei CO%
6. Notieren Sie den Wert PV = _____
7. Geben Sie ein: `60` bei CO%
8. Notieren Sie den Wert PV = _____
9. Geben Sie ein: `80` bei CO%
10. Notieren Sie den Wert PV = _____
11. Die notierten Werte sollten von CO% um denselben Betrag versetzt sein. Dies beweist, dass der Prozess linear ist. Das folgende Beispiel zeigt einen Versatz um den Wert 15.
    - CO 20% = PV 35%
    - CO 40% = PV 55%
    - CO 60% = PV 75%
    - CO 80% = PV 95%

Wenn die notierten Werte nicht um den gleichen Betrag versetzt sind, liegt eine der folgenden Situationen vor:

- die Skalierung ist nicht korrekt
- der Prozess ist nicht linear
- die Geräte sind nicht ordnungsgemäß angeschlossen und/oder konfiguriert

Ergreifen Sie die notwendigen Abhilfemaßnahmen, und wiederholen Sie anschließend die Schritte 2-10.

## Bestimmen der ersten Regelkreisaktualisierungszeit

Um die genäherte Regelkreisaktualisierungszeit für den jeweiligen Prozess festzustellen, führen Sie folgende Schritte durch:

1. Geben Sie unter MinS und MaxS die normalen Anwendungswerte ein.
2. Geben Sie ein: `50` bei CO%
3. Geben Sie ein: `60` bei CO%, und starten Sie dann sofort eine Stoppuhr.
4. Überwachen Sie den Ausgang PV. Stoppen Sie die Zeit, wenn der PV-Wert beginnt, sich zu verändern. Notieren Sie den gemessenen Wert. Dies ist die Totzeit.
5. Multiplizieren Sie die Totzeit mit 4. Dieser Wert entspricht etwa der natürlichen Schwingungsdauer. Wenn beispielsweise die Totzeit = 3 Sekunden, dann ergibt sich 4 x 3 = 12 Sekunden (≅ natürliche Schwingungsdauer).
6. Teilen Sie den in Schritt 5 errechneten Wert durch 10. Verwenden Sie diesen Wert als Regelkreisaktualisierungszeit. Beispiel:

   Natürliche Schwingungsdauer = 12 Sekunden, dann 12/10 = 1,2 Sekunden.

   Als Regelkreisaktualisierungszeit sollte daher der Wert 120 eingegeben werden.
   (120 x 10 ms = 1,2 Sekunden)
7. Geben Sie die folgenden Werte ein: den Anfangs-Sollwert SP, die Integralzeit $T_i$ 0, die Differenzialzeit $T_d$ 0, den Verstärkungsfaktor $K_c$ 1 und die in Schritt 17 ermittelte Regelkreisaktualisierungszeit.

   Wählen Sie im Kontaktplan den STI-Modus oder den Zeitsteuerungsmodus aus. Achten Sie bei der Wahl von STI darauf, dass die Regelkreisaktualisierungszeit dem STI-Zeitintervall entspricht.

   Geben Sie die gewünschten optionalen Einstellungen ein (Ausgangsbegrenzung, Ausgangsalarm, Smax - Smin-Skalierung, Störgrößenaufschaltung).
8. Kehren Sie auf Seite 19-24 zurück, und setzen Sie die Arbeits- schritte zur Abstimmung beginnend bei Schritt 4 fort.

# ASCII-Befehle

Dieses Kapitel enthält generelle Informationen zu den ASCII-Befehlen sowie deren Funktionsweise im Steuerprogramm. Folgende Befehle werden in diesem Kapitel beschrieben:

## Allgemeine Informationen


- „Befehlstypen und -funktionsweise“ auf Seite 20-2
- „Übersicht zu den Protokollen“ auf Seite 20-4
- „ST-Datenfile (String; Zeichenkette)“ auf Seite 20-5
- „Steuerdatenfile“ auf Seite 20-6


## ASCII-Befehle

Die ASCII-Befehle sind so angeordnet, dass die Schreibbefehle vor den Lesebefehlen stehen.

| Befehl | Funktion | Gültige Steuerung(en) | Seite |
|---|---|---|---|
| ACL - ASCII-Puffer löschen | Den Empfangs- und/oder den Sendepuffer löschen. | • MicroLogix 1200<br>• MicroLogix 1500 Serie B, FRN 4 oder höher | 20-7 |
| AIC - Ganzzahl in Zeichenkette | Einen ganzzahligen Wert in eine Zeichenkette umwandeln. | | 20-8 |
| AWA - ASCII schreiben und anhängen | Eine Zeichenkette mit Anhang mit anwenderkonfigurierten Zeichen schreiben. | | 20-9 |
| AWT - ASCII schreiben | Eine Zeichenkette schreiben. | | 20-12 |
| ABL - Puffer auf Zeile überprüfen | Die Zeichenanzahl im Puffer bestimmen, einschließlich des Zeilenendezeichens. | • MicroLogix 1200 Serie B, FRN 3 oder höher<br>• MicroLogix 1500 Serie B, FRN 4 oder höher | 20-14 |
| ACB - Anzahl der ASCII-Zeichen im Puffer | Die Gesamtzahl der Zeichen in Puffer bestimmen. | | 20-16 |
| ACI - Zeichenkette in Ganzzahl | Eine Zeichenkette in einen ganzzahligen Wert umwandeln. | | 20-17 |
| ACN - Zeichenkette verketten | Zwei Zeichenketten zu einer verbinden. | | 20-19 |
| AEX - Zeichenkette extrahieren | Einen Teil einer Zeichenkette extrahieren, um eine neue Zeichenkette zu erzeugen. | | 20-20 |
| AHL - ASCII-Handshake-Leitungen | Modem-Handshake-Leitungen setzen oder zurücksetzen. | | 20-21 |
| ARD - ASCII-Lesen | Zeichen aus dem Eingangspuffer lesen und diese in einer Zeichenkette platzieren. | | 20-23 |
| ARL - ASCII-Zeile lesen | Eine Zeile aus dem Eingangspuffer lesen und diese in einer Zeichenkette platzieren. | | 20-24 |
| ASC - Zeichenkette suchen | Zeichenkette suchen. | | 20-26 |
| ASR - ASCII-Zeichenkette vergleichen | Zwei Zeichenketten vergleichen. | | 20-28 |

## Befehlstypen und -funktionsweise

Generell werden zwei ASCII-Befehlstypen unterschieden, ASCII- Befehle für die Zeichenkettensteuerung und ASCII-Befehle für die Anschlusssteuerung. Der ASCII-Befehl für die Zeichenketten- steuerung wird sofort ausgeführt und dient zur Bearbeitung von Daten. Der ASCII-Befehl für die Anschlusssteuerung wird zur Übertragung von Daten verwendet und benutzt dabei die ASCII- Warteschlange. Weitere Einzelheiten finden Sie nachfolgend.

### ASCII-Zeichenkettensteuerung

Diese Befehle werden zur Bearbeitung von Zeichenketten verwendet. Erkennt das System einen Zeichenketten-Steuerungsbefehl in einem Kontaktplanlogikprogramm, wird dieser sofort ausgeführt. Dieser Befehl wird niemals in eine Warteschlange eingereiht. Die folgenden Tabellen enthalten die ASCII-Zeichenketten-Steuerungsbefehle, die von den MicroLogix 1200- und 1500-Steuerungen verwendet werden:

| **MicroLogix 1200 Serie A** |
|---|
| AIC (Ganzzahl in Zeichenkette) |

| **MicroLogix 1200 Serie B, FRN 3 und höher**<br>**MicroLogix 1500 Serie B, FRN 4 und höher** | |
|---|---|
| ACI (Zeichenkette in Ganzzahl) | AIC (Ganzzahl in Zeichenkette) |
| ACN (Zeichenkette verketten) | ASC (Zeichenkette suchen) |
| AEX (Zeichenkette extrahieren) | ASR (ASCII-Zeichenkette vergleichen) |

### ASCII-Anschlusssteuerung

Diese Befehle verwenden oder ändern den Kommunikationskanal zum Empfang oder zur Übertragung von Daten. Die folgenden Tabellen enthalten die ASCII-Anschluss-Steuerungsbefehle, die von den MicroLogix 1200- und 1500-Steuerungen verwendet werden:

| **MicroLogix 1200 Serie A(1)** |
|---|
| ACL (ASCII-Puffer löschen) |
| AWA (ASCII schreiben und anhängen) |
| AWT (ASCII schreiben) |

(1) Bei den MicroLogix 1200-Steuerungen der Serie A dienen diese Befehle ausschließlich zur Datenübertragung.

| **MicroLogix 1200 Serie B, FRN 3 und höher**<br>**MicroLogix 1500 Serie B, FRN 4 und höher** | |
|---|---|
| ABL (Puffer auf Zeile überprüfen) | ARD (ASCII-Lesen) |
| ACB (Anzahl der ASCII-Zeichen im Puffer) | ARL (ASCII-Zeile lesen) |
| ACL (ASCII-Puffer löschen) | AWA (ASCII schreiben und anhängen) |
| AHL (ASCII-Handshake-Leitungen) | AWT (ASCII schreiben) |

Erkennt das System den ACL-Befehl (ASCII Clear Buffer) in einem Kontaktplanlogik-Programm, wird dieser sofort ausgeführt. Dies hat zur Folge, dass alle Befehle aus der ASCII-Warteschlange gelöscht werden (dabei wird auch der aktuell ausgeführte ASCII-Befehl gestoppt). Für jeden aus der ASCII-Warteschlange gelöschten Befehl wird das Fehler-Bit (ER) gesetzt.

Wenn ein beliebiger anderer Anschluss-Steuerungsbefehl in einem Kontaktplanlogik-Programm erkannt wird, kann er – abhängig von den Inhalten der ASCII-Warteschlange – sofort ausgeführt werden oder nicht. Die

ASCII-Warteschlange basiert auf dem FIFO-Prinzip (First-In, First-Out) und hat eine Kapazität von bis zu 16 Befehlen. Funktionsweise der ASCII-Warteschlange:

- Wenn ein Befehl in einem Strompfad vorkommt und die ASCII-Warteschlange leer ist, wird der Befehl sofort ausgeführt. Unter Umständen vergehen mehrere Programmabfragen, bis der Befehl abgeschlossen ist.
- Erkennt das System den Befehl in einem Strompfad, wenn sich 1 bis 15 Befehle in der ASCII-Warteschlange befinden, wird der Befehl in die Warteschlange gestellt und nach Ausführung der vorhergehenden Befehle ausgeführt. Ist die ASCII-Warteschlange voll, wartet der Befehl auf die nächste Programmabfrage, um festzustellen, ob dann Platz in der ASCII-Warteschlange ist. Die Steuerung fährt mit der Ausführung der anderen Befehle fort, während der ASCII-Anschluss-Steuerungsbefehl auf einen freien Platz in der Warteschlange wartet.

## Programmieren von ASCII-Befehlen

Beim Programmieren von ASCII-Ausgangsbefehlen muss dem ASCII- Befehl stets eine bedingte Logik vorangestellt werden, die erkennt, wann neue Daten gesendet werden müssen. Alternativ hierzu können Sie Daten auch in einem bestimmten Zeitintervall senden. In diesem Fall muss ein Intervall von mindestens 0,5 Sekunden verwendet werden. Generieren Sie keine kontinuierlichen Ströme von ASCII- Daten aus einem Kommunikationsanschluss.

**WICHTIG** Werden ASCII-Schreibbefehle kontinuierlich ausgeführt, können Sie eventuell die Kommunikation mit RSLogix 500 nicht wiederherstellen, wenn die Steuerung in den Run-Modus wechselt.

# Übersicht zu den Protokollen

## MicroLogix 1200 Serie A und MicroLogix 1500 Serie A

Die AWA- und AWT-Befehle übertragen nur dann einen ASCII-Befehl aus dem RS-232-Anschluss, wenn der Kanal für das DF1-Vollduplex- Protokoll konfiguriert ist. Ist der RS-232-Anschluss für ein anderes Protokoll (also nicht DF1-Vollduplex) konfiguriert, werden die AWA- und AWT-Befehle mit einem Fehlercode von 9 beendet.

DF1 Vollduplex-Pakete haben Vorrang vor ASCII-Zeichenketten, d. h. wenn ein AWA- oder ein AWT-Befehl während der Übertragung eines F1 Vollduplex-Pakets ausgelöst wird, wird für den ASCII-Befehl ein Fehler mit Fehlercode 5 generiert.

Siehe Tabelle E.2 auf Seite E-5. Diese Tabelle enthält die DF1-Vollduplex-Protokollparamter, die Sie im Konfigurationsbildschirm *Channel 0 (Kanal 0)* Ihrer Programmiersoftware festlegen können. Die Konfiguration der beiden angehängten Zeichen für den AWA- Befehl erfolgt in RSLogix 500 über die Auswahloption *Channel Configuration (Kanalkonfiguration)* auf der Registerkarte *General (Allgemein)*.

## MicroLogix 1200 Serie B, FRN 3 und höher sowie MicroLogix 1500 Serie B, FRN 4 und höher

Für die AWA- und AWT-Befehle kann das DF1-Vollduplex-Protokoll wie oben beschrieben verwendet werden. Wenn Sie den gesamten ASCII-Befehlssatz nutzen möchten, verwenden Sie das ASCII-Protokoll wie im Folgenden beschrieben.

Siehe Tabelle E.9 auf Seite E-16. Diese Tabelle enthält die ASCII- Parameter, die Sie in den Konfigurationsbildschirmen *Kanal 0* (und *Kanal 1* bei 1764-LRP) Ihrer Programmiersoftware festlegen können. Die Konfiguration der beiden angehängten Zeichen für den AWA-Befehl erfolgt in RSLogix 500 über die Auswahloption *Channel Configuration (Kanalkonfiguration)* auf der Registerkarte *General (Allgemein)*.

# ST-Datenfile (String; Zeichenkette)

## Filebeschreibung

In dem ST-Datenfile werden von den ASCII-Befehlen ASCII- Zeichendaten gespeichert. Der Zugriff auf die ASCII-Daten erfolgt durch die Quellen- und Zieloperanden in den ASCII-Befehlen. Der ST-Datenfile kann auch von dem Kopierbefehl (COP) und den Verschiebebefehlen (MOV, MVM) verwendet werden.

ST-Files bestehen aus 42-Wort-Elementen. Ein ST-Fileelement ist nachfolgend dargestellt. In einem ST-File können sich bis zu 256 dieser Elemente befinden.

**Tabelle 20.1 Struktur der Zeichenketten-Datenfiles**

| | **Zeichenkettenelement** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | **15** | **14** | **13** | **12** | **11** | **10** | **09** | **08** | **07** | **06** | **05** | **04** | **03** | **02** | **01** | **00** |
| **Wort** | ***höherwertiges Byte*** | | | | | | | | ***niederwertiges Byte*** | | | | | | | |
| 0 | Zeichenkettenlänge - Anzahl der Zeichen (der Bereich liegt zwischen 0 und 82) | | | | | | | | | | | | | | | |
| 1 | Zeichen 0 | | | | | | | | Zeichen 1 | | | | | | | |
| 2 | Zeichen 2 | | | | | | | | Zeichen 3 | | | | | | | |
| ↓ | ↓ | | | | | | | | ↓ | | | | | | | |
| 40 | Zeichen 78 | | | | | | | | Zeichen 79 | | | | | | | |
| 41 | Zeichen 80 | | | | | | | | Zeichen 81 | | | | | | | |

## Adressierung von ST-Files

Die Adressierung von ST-Datenfiles wird nachfolgend beschrieben.

| **Format** | **Bedeutung** | | |
|---|---|---|---|
| | **ST** | Zeichenketten-File | |
| **STf:e.s** | **f** | Filenummer | Die gültige Filenummer kann aus einem Bereich von 3 bis 255 ausgewählt werden. |
| | **:** | Elementendezeichen | |
| | **e** | Elementnummer | Die Elementnummer kann aus einem Bereich von 0 bis 255 ausgewählt werden.<br>Jedes Element hat eine Länge von 42 Worten (siehe Tabelle 20.1). |
| | **.** | Subelementendezeichen | |
| | **s** | Subelementnummer | Die gültige Subelementnummer kann aus einem Bereich von 0 bis 41 ausgewählt werden. Das Wort 0 kann außerdem mit .LEN bezeichnet werden.<br>Das Subelement steht für eine Wortadresse. |
| **Beispiele:** | **ST9:2**<br>**ST17:1.LEN** | | **Zeichenkettenfile 9, Element 2**<br>**Zeichenkettenfile 17, Element 1, LEN-Variable** |

## Steuerdatenfile

### Filebeschreibung

Das Steuerdatenelement wird vom ASCII-Befehl verwendet, um die Steuerdaten zu speichern, die zum Ausführen des Befehls erforderlich sind. Das Steuerdatenelement für ASCII-Befehle enthält Status- und Steuer-Bits, ein Fehlercode-Byte und aus zwei Zeichen bestehende Worte (siehe unten):

**Tabelle 20.2 Steuerdaten-Fileelemente für ASCII-Befehle**

| | **Steuerelement** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort** | **15** | **14** | **13** | **12** | **11** | **10** | **09** | **08** | **07** | **06** | **05** | **04** | **03** | **02** | **01** | **00** |
| 0 | EN(1) | EU(2) | DN(3) | EM(4) | ER(5) | UL(6) | RN(7) | FD(8) | Fehlercodebyte | | | | | | | |
| 1 | Angegebene Anzahl der zu sendenden oder empfangenden Zeichen (LEN) | | | | | | | | | | | | | | | |
| 2 | Anzahl der tatsächlich gesendeten/empfangenen Zeichen (POS) | | | | | | | | | | | | | | | |

(1) EN - Das Freigabe-Bit wird bei einem Übergang des Strompfades von unwahr nach wahr gesetzt und zeigt an, dass der Befehl aktiviert wurde. Dieses Bit bleibt gesetzt, bis die Ausführung des Befehls abgeschlossen ist oder bei dieser Ausführung ein Fehler generiert wurde.

(2) EU = Warteschlangen-Bit - Ist dieses gesetzt, weist es darauf hin, dass ein ASCII-Befehl in die ASCII-Warteschlange eingereiht wurde. Dieser Vorgang wird verzögert, wenn die Warteschlange bereits voll ist.

(3) DN = Asynchrones Fertig-Bit - wird gesetzt, wenn die Durchführung eines Befehls erfolgreich abgeschlossen wurde.

(4) EM = Synchrones Fertig-Bit - nicht verwendet

(5) ER = Fehler-Bit - ist dieses gesetzt, weist es darauf hin, dass während der Ausführung des Befehls ein Fehler auftrat.

(6) UL = Entlade-Bit - setzt der Anwender dieses Bit, wird der Befehl nicht ausgeführt. Führt das System den Befehl bereits aus, wird die Funktion unterbrochen. Wird dieses Bit während der Ausführung eines Befehls gesetzt, werden alle bereits verarbeiteten Daten an die Zieladresse gesendet. Alle übrigen Daten werden nicht verarbeitet. Durch Setzen dieses Bits werden keine Befehle aus der ASCII-Warteschlange gelöscht. Dieses Bit wird nur geprüft, wenn der Befehl zur Ausführung bereit ist.

(7) RN = Ausführungs-Bit - ist dieses gesetzt, weist es darauf hin, dass der Befehl in der Warteschlange ausgeführt wird.

Hinweis: Das Ausführungs-Bit (RN) kann nicht über den Control (R)-File adressiert werden.

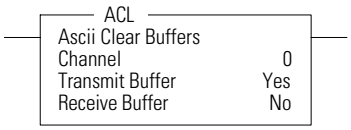
(8) FD = Gefunden-Bit - ist dieses gesetzt, weist es darauf hin, dass der Befehl die letzte Zeile oder das Abschlusszeichen im Puffer gefunden hat. (wird nur von den ABL- und ACB-Befehlen verwendet)

### Adressierung von Steuerungsfiles

Die Adressierung von Steuerdatenfiles wird nachfolgend beschrieben.

| **Format** | **Bedeutung** | | |
|---|---|---|---|
| | **R** | Steuerungsfile | |
| **R:e.s/b** | **f** | Filenummer | Die gültige Filenummer kann aus einem Bereich von 3 bis 255 ausgewählt werden. |
| | **:** | Elementendezeichen | |
| | **e** | Elementnummer | Die Elementnummer kann aus einem Bereich von 0 bis 255 ausgewählt werden.<br>Jedes Element hat eine Länge von 3 Worten (siehe Tabelle 20.2). |
| | **.** | Subelementendezeichen | |
| | **s** | Subelementnummer | Die gültige Subelementnummer kann aus einem Bereich von 0 bis 2 ausgewählt werden.<br>Außerdem kann .LEN oder .POS angegeben werden. |
| | **/** | Bit-Endezeichen | |
| | **b** | Bitnummer | Die Bitnummer kann aus einem Bereich von 0 bis 15 ausgewählt werden.<br>Die Bitnummer ist der Bitstandort innerhalb des ST-Fileelements.<br>Die Wörter 1 und 2 der Steuerungselemente können nicht auf Bitebene adressiert werden. |
| **Beispiele:** | **R6:2**<br>**R6:2.0/13**<br>**R18:1.LEN**<br>**R18:1.POS** | | **Element 2, Steuerfile 6**<br>**Bit 13 in Subelement 0 von Element 2, Steuerungfile 6**<br>**Angegebene Zeichenkettenlänge von Element 1, Steuerungsfile 18**<br>**Tatsächliche Zeichenkettenlänge von Element 1, Steuerungsfile 18** |

# ACL – ASCII-Puffer löschen




Befehlstyp: Ausgang

**Tabelle 20.3 Ausführungszeit des ACL-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | **wahr** | **unwahr** |
| MicroLogix 1200 | Puffer löschen:<br>beide 249,1 µs<br>empfangen 28,9 µs<br>übertragen 33,6 µs | 0,0 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | Puffer löschen:<br>beide 203,9 µs<br>empfangen 24,7 µs<br>übertragen 29,1 µs | 0,0 µs |

Der ACL-Befehl löscht die Empfangs- und/oder Übertragungspuffer. Dieser Befehl löscht auch Befehle aus der ASCII-Warteschlange.

Dieser Befehl wird unmittelbar im Anschluss an den Übergang des Strompfads von unwahr nach wahr ausgeführt. Alle laufenden ASCII-Übertragungen werden bei Ausführung des ACL-Befehls beendet.

**HINWEIS** Die ASCII-Warteschlange kann bis zu 16 Befehle enthalten, die auf ihre Ausführung warten.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Channel (Kanal)** ist die Nummer des RS-232-Anschlusses, Kanal 0. (*Nur 1764-LRP: Auswahl von Kanal 0 oder Kanal 1 möglich.*)
- **Receive Buffer (Empfangspuffer)** löscht den Empfangspuffer, wenn dieser Parameter auf „Yes“ (Ja) gesetzt ist. Er löscht die Steuerungsbefehle für „ASCII-Anschluss empfangen“ (ARL und ARD) aus der ASCII-Warteschlange.
- **Transmit Buffer (Übertragungspuffer)** löscht den Übertragungspuffer, wenn dieser Parameter auf „Yes“ (Ja) gesetzt ist. Er löscht die Steuerungsbefehle für „ASCII-Anschluss übertragen“ (AWA und AWT) aus der ASCII-Warteschlange.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.4 Gültige Adressierungsmodi und Filetypen für ACL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles(1) | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Kanal | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Empfangspuffer | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Übertragungspuffer | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |

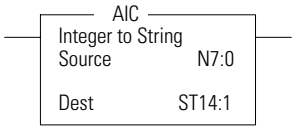
(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

## Funktionsweise

Wenn „Clear Receive Buffer" (Empfangspuffer löschen) und „Clear Transmit Buffer" (Übertragungspuffer löschen) beide auf „Yes" (Ja) gesetzt sind, werden alle Empfangs- und Übertragungsbefehle (ARL, ARD, AWA und AWT) aus der ASCII-Warteschlange entfernt.

Beim Löschen der Befehle aus der ASCII-Warteschlange werden die folgenden Bits gesetzt: ER = 1, RN = 0, EU = 0 und ERR = 0x0E.

# AIC – ASCII-Ganzzahl in Zeichenkette




Befehlstyp: Ausgang

**Tabelle 20.5 Ausführungszeit des AIC-Befehls**

| Steuerung | Datengröße | Befehl | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | Wort | 29,3 µs + 5,2 µs/Zeichen | 0,0 µs |
| | Doppelwort | 82,0 µs | 0,0 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | Wort | 25 µs + 4,3 µs/Zeichen | 0,0 µs |
| | Doppelwort | 68,7 µs | 0,0 µs |

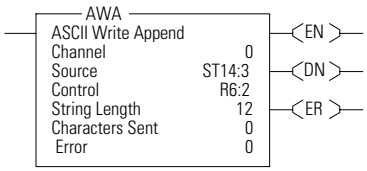
Mit dem ACIBefehl wird ein Ganzzahl- oder Doppelwortwert (Quelle) in eine ASCIIZeichenkette (Ziel) umgewandelt. Die Quelle kann eine Konstante oder eine Adresse sein. Die Quellendaten liegen in einem Bereich von -2.147.483.648 bis 2.147.483.647.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.6 Gültige Adressierungsmodi und Filetypen für AIC-Befehl**
*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | | • | | | | | | | | | | | | | | | • | • | | | • | • | |
| Ziel | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |

# AWA – ASCII schreiben und anhängen




Befehlstyp: Ausgang

**Tabelle 20.7 Ausführungszeit des AWA-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 268 µs + 12 µs/Zeichen | 14,1 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 236 µs + 10,6 µs/Zeichen | 12,5 µs |

Mit dem AWA-Befehl können Sie Zeichen aus einer Quellenzeichen- kette an ein externes Gerät schreiben. Dabei werden die beiden angehängten Zeichen hinzugefügt, die im Konfigurationsfenster für den Kanal festgelegt wurden. Als Standardeinstellung werden an das Ende der Zeichenkette ein Wagenrücklauf- und ein Zeilenvorschub- zeichen angehängt.

**HINWEIS** Die angehängten Zeichen werden im Konfigurations- fenster für den Kanal festgelegt. Als Standardeinstellung werden ein Wagenrücklauf- und ein Zeilenvorschubzeichen angehängt.

## Programmieren von AWA-Befehlen

Beim Programmieren von ASCII-Ausgangsbefehlen muss dem ASCII-Befehl stets eine bedingte Logik vorangestellt werden, die erkennt, wann neue Daten gesendet werden müssen. Alternativ hierzu können Sie Daten auch in einem bestimmten Zeitintervall senden. In diesem Fall muss ein Intervall von mindestens 0,5 Sekunden verwendet werden. Generieren Sie keine kontinuierlichen Ströme von ASCII-
Daten aus einem Kommunikationsanschluss.

**WICHTIG** Werden ASCII-Schreibbefehle kontinuierlich ausgeführt, können Sie eventuell die Kommunikation mit RSLogix 500 nicht wiederherstellen, wenn die Steuerung in den Run-Modus wechselt.

Dieser Befehl wird auf wahren und unwahren Strompfaden ausge- führt. Wenn dieser Befehl jedoch wiederholt werden soll, muss der Strompfad von unwahr nach wahr übergehen.

Bei diesem Befehl können Sie auch die eingeschleifte indirekte Adres- sierung verwenden. Weitere Informationen hierzu finden Sie auf Seite 20-30.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Channel (Kanal)** ist die Nummer des RS-232-Anschlusses, Kanal 0. (*Nur 1764-LRP: Auswahl von Kanal 0 oder Kanal 1 möglich.*).
- **Source (Quelle)** ist das Zeichenkettenelement, das geschrieben werden soll.
- **Control (Steuerung)** ist der Steuerdatenfile. Siehe Seite 20-6.
- **String Length (Zeichenkettenlänge)** (.LEN) ist die Anzahl der aus der Quellenzeichenkette zu schreibenden Zeichen (0 bis 82). Bei Eingabe von 0 wird die gesamte Zeichenkette geschrieben. Dies ist Wort 1 in dem Steuerdatenfile.
- **Characters Sent (Gesendete Zeichen)** (.POS) ist die Anzahl der Zeichen, die an ein externes Gerät gesendet werden. Dies ist Wort 2 in dem Steuerdatenfile. Gesendete Zeichen (.POS) wird nach Übertragung aller Zeichen aktualisiert.

  Der gültige Bereich für .POS liegt zwischen 0 und 84. Die Anzahl der Zeichen, die an das Ziel gesendet wurden, kann kleiner oder größer als die angegebene Zeichenkettenlänge (.LEN) sein (siehe unten):
  - Die Anzahl der gesendeten Zeichen (.POS) kann kleiner sein als die Zeichenkettenlänge (.LEN), wenn die Länge der tatsächlich gesendeten Zeichenlänge kleiner ist als die in dem Feld Zeichenkettenlänge (.LEN) angegebene Zahl.
  - Die Anzahl der gesendeten Zeichen (.POS) kann größer sein als die Zeichenkettenlänge (.LEN), wenn die angehängten Zeichen oder eingefügte Werte aus der eingeschleiften indirek- ten Adressierung verwendet werden. Bei einer Zeichen- kettenlänge (.LEN) von mehr als 82 wird die Zeichenkette, die an das Ziel geschrieben wird, auf 82 Zeichen plus der Anzahl der angehängten Zeichen gekürzt (also je nach Anzahl der angehängten Zeichen auf insgesamt 82, 83 oder 84 Zeichen).
- **Error (Fehler)** zeigt den hexadezimalen Fehlercode an, der angibt, warum das ER-Bit im Datenfile gesetzt wurde. Eine Beschreibung der Fehlercodes finden Sie auf Seite 20-31.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

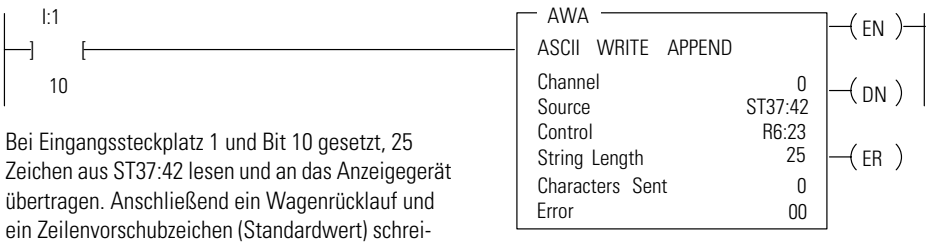
**Tabelle 20.8 Gültige Adressierungsmodi und Filetypen für AWA-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Kanal | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Quelle | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Steuerung | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

*Beispiel:*




Bei Eingangssteckplatz 1 und Bit 10 gesetzt, 25 Zeichen aus ST37:42 lesen und an das Anzeigegerät übertragen. Anschließend ein Wagenrücklauf und ein Zeilenvorschubzeichen (Standardwert) schreiben.

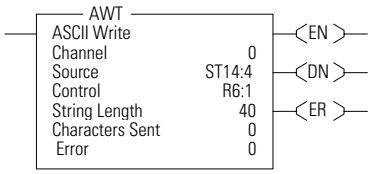
In diesem Beispiel wird bei einem Übergang des Strompfads von unwahr nach wahr das Freigabe-Bit (EN) des Steuerungselements gesetzt. Wenn der Befehl in die ASCII-Warteschlange gestellt wird, wird das Warteschlangen-Bit (EU) gesetzt. Das Ausführungs-Bit (RN) wird gesetzt, wenn der Befehl ausgeführt wird.Das DN-Bit wird gesetzt, wenn der Befehl abgeschlossen ist.

Die Steuerung sendet 25 Zeichen vom Anfang der Zeichenkette ST37:42 an das Anzeigegerät. Anschließend sendet es anwenderkon- figurierte angehängte Zeichen. Das Fertig-Bit (DN) wird gesetzt und der Wert 27 in das .POS-Wort des ASCII-Steuerdatenfiles gestellt.

Wenn ein Fehler auftritt, werden der Fehlercode in das Fehlercode- byte geschrieben und das Fehler-Bit (ER) gesetzt. Eine Liste der Fehlercodes und der entsprechenden Maßnahmen finden Sie im Abschnitt „Fehlercodes zu ASCII-Befehlen" auf Seite 20-31.

**HINWEIS** Informationen zur zeitlichen Abstimmung dieses Befehls finden Sie im Zeitdiagramm auf Seite 20-29.

# AWT – ASCII schreiben




Befehlstyp: Ausgang

**Tabelle 20.9 Ausführungszeit des AWT-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | **wahr** | **unwahr** |
| MicroLogix 1200 | 268 µs + 12 µs/Zeichen | 14,1 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 237 µs + 10,6 µs/Zeichen | 12,8 µs |

Mit dem AWT-Befehl können Sie Zeichen aus einer Quellenzeichenkette an ein externes Gerät schreiben.

## Programmieren von AWT-Befehlen

Beim Programmieren von ASCII-Ausgangsbefehlen ist dem ASCII-Befehl stets eine bedingte Logik voranzustellen, die erkennt, wann neue Daten gesendet werden müssen. Alternativ hierzu können Sie Daten auch in einem bestimmten Zeitintervall senden. In diesem Fall muss ein Intervall von mindestens 0,5 Sekunden verwendet werden.

**WICHTIG** Generieren Sie keine kontinuierlichen Ströme von ASCII-Daten aus einem Kommunikationsanschluss. Werden ASCII-Schreibbefehle kontinuierlich ausgeführt, können Sie eventuell die Kommunikation mit RSLogix 500 nicht wiederherstellen, wenn die Steuerung in den Run-Modus wechselt.

Dieser Befehl wird auf einem wahren Strompfad ausgeführt. Wird der Strompfad nach dem Start unwahr, führt das System den Befehl den- noch aus. Soll dieser Befehl wiederholt werden, muss der Strompfad von unwahr nach wahr übergehen.

Bei diesem Befehl können Sie auch die eingeschleifte indirekte Adres- sierung verwenden. Weitere Informationen hierzu finden Sie auf Seite 20-30.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Channel (Kanal)** ist die Nummer des RS-232-Anschlusses, Kanal 0. (*Nur 1764-LRP: Auswahl von Kanal 0 oder Kanal 1 möglich.*).
- **Source (Quelle)** ist das Zeichenkettenelement, das geschrieben werden soll.
- **Control (Steuerung)** ist der Steuerdatenfile. Siehe Seite 20-6.
- **String Length (Zeichenkettenlänge)** (.LEN) ist die Anzahl der aus der Quellenzeichenkette zu schreibenden Zeichen (0 bis 82). Bei Eingabe von 0 wird die gesamte Zeichenkette geschrieben. Dies ist Wort 1 in dem Steuerdatenfile.
- **Characters Sent (Gesendete Zeichen)** (.POS) ist die Anzahl der Zeichen, die an ein externes Gerät gesendet werden. Dies ist Wort 2 in dem Steuerdatenfile. Gesendete Zeichen (.POS) wird nach Übertragung aller Zeichen aktualisiert.

  Der gültige Bereich für .POS liegt zwischen 0 und 82. Die Anzahl der Zeichen, die an das Ziel gesendet wurden, kann kleiner oder größer als die angegebene Zeichenkettenlänge (.LEN) sein (siehe unten):
  - Die Anzahl der gesendeten Zeichen (.POS) kann kleiner sein als die Zeichenkettenlänge (.LEN), wenn die Länge der tatsächlich gesendeten Zeichenlänge kleiner ist als die in dem Feld Zeichenkettenlänge (.LEN) angegebene Zahl.
  - Die Anzahl der gesendeten Zeichen (.POS) kann größer sein als die Zeichenkettenlänge (.LEN), wenn eingefügte Werte aus der eingeschleiften indirekten Adressierung verwendet wer- den. Wenn die Zeichenkettenlänge (.LEN) größer als 82 ist, wird die Zeichenkette an das Ziel auf 82 Zeichen gekürzt.
- **Error (Fehler)** zeigt den hexadezimalen Fehlercode an, der angibt, warum das ER-Bit im Datenfile gesetzt wurde. Eine Beschreibung der Fehlercodes finden Sie auf Seite 20-31.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

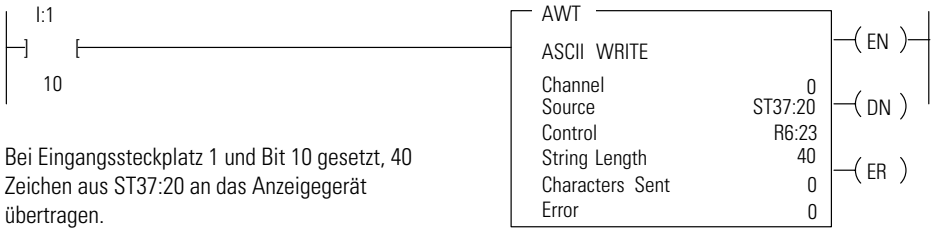
**Tabelle 20.10 Gültige Adressierungsmodi und Filetypen für AWT-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Kanal | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Quelle | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Steuerung | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

*Beispiel:*




Bei Eingangssteckplatz 1 und Bit 10 gesetzt, 40 Zeichen aus ST37:20 an das Anzeigegerät übertragen.

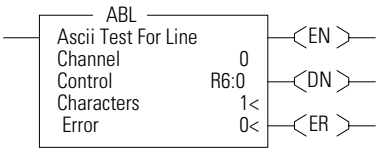
In diesem Beispiel wird bei einem Übergang des Strompfads von unwahr nach wahr das Freigabe-Bit (EN) des Steuerungselements gesetzt. Wenn der Befehl in die ASCII-Warteschlange gestellt wird, wird das Warteschlangen-Bit (EU) gesetzt. Das Ausführungs-Bit (RN) wird gesetzt, wenn der Befehl ausgeführt wird. Das DN-Bit wird gesetzt, wenn der Befehl abgeschlossen ist.

Es werden 40 Zeichen aus Zeichenkette ST37:40 über Kanal 0 versendet. Das Fertig-Bit (DN) wird gesetzt und der Wert 40 in das .POS-Wort des ASCII-Steuerdatenfiles gestellt.

Wenn ein Fehler auftritt, werden der Fehlercode in das Fehlercode- byte geschrieben und das Fehler-Bit (ER) gesetzt. Eine Liste der Fehlercodes und der entsprechenden Maßnahmen finden Sie im Abschnitt „Fehlercodes zu ASCII-Befehlen“ auf Seite 20-31.

**HINWEIS** Informationen zur zeitlichen Abstimmung dieses Befehls finden Sie im Zeitdiagramm auf Seite 20-29.

## ABL – Puffer auf Zeile überprüfen




Befehlstyp: Ausgang

**Tabelle 20.11 Ausführungszeit des ABL-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | 115 µs + 8,6 µs/Zeichen | 12,5 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 94 µs + 7,6 µs/Zeichen | 11,4 µs |

Der ABL-Befehl dient zum Bestimmen der Zeichenanzahl im Empfangspuffer des angegebenen Kommunikationskanals (bis ein- schließlich der ersten Zeilenendezeichen bzw. Abschlusszeichen). Dabei sucht der Befehl nach den beiden Abschlusszeichen, die über das Konfigurationsfenster für den Kanal festgelegt wurden. Beim Übergang von unwahr nach wahr meldet die Steuerung die Anzahl der Zeichen im POS-Feld des Steuerdatenfiles. Der Kanal muss für ASCII konfiguriert sein.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Channel (Kanal)** ist die Nummer des RS-232-Anschlusses, Kanal 0. (*Nur 1764-LRP: Auswahl von Kanal 0 oder Kanal 1 möglich.*).
- **Control (Steuerung)** ist der Steuerdatenfile. Siehe Seite 20-6.
- **Characters (Zeichen)** ist die Anzahl der von der Steuerung erkannten Zeichen im Puffer (0 bis 1024). Dieser Parameter ist schreibgeschützt und befindet sich in Wort 2 des Steuerdatenfiles.
- **Error (Fehler)** zeigt den hexadezimalen Fehlercode an, der angibt, warum das ER-Bit im Datenfile gesetzt wurde. Eine Beschreibung der Fehlercodes finden Sie auf Seite 20-31.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.12 Gültige Adressierungsmodi und Filetypen für ABL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Kanal | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Steuerung | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |

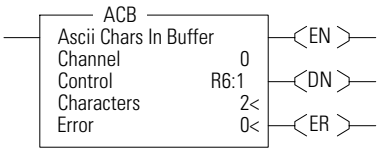
(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

## Funktionsweise

Wenn der Zustand des Strompfads von unwahr nach wahr übergeht, wird das EN-Bit gesetzt. Der Befehl wird in die ASCII-Befehlswarte- schlange eingereiht, das Warteschleifen-Bit (EU) wird gesetzt, und die Programmabfrage wird fortgesetzt. Anschließend wird der Befehl außerhalb der Programmabfrage ausgeführt. Ist allerdings die Warteschlange leer, wird der Befehl sofort ausgeführt. Bei der Ausführung wird das Ausführungs-Bit (RN) gesetzt.

Die Steuerung bestimmt die Anzahl der Zeichen (bis einschließlich der ersten Abschlusszeichen) und fügt diesen Wert in das POS-Feld des Steuerdatenfiles ein. Anschließend wird das Fertig-Bit (DN) gesetzt. Erscheint im POS-Feld eine Null, wurde kein Abschlusszeichen gefun- den. Das Gefunden-Bit (FD) wird gesetzt, wenn das POS-Feld auf einen Wert ungleich Null gesetzt wurde.

# ACB – Anzahl der ASCII-Zeichen im Puffer




Befehlstyp: Ausgang

**Tabelle 20.13 Ausführungszeit des ACB-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | 103.1 | 12,1 |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 84,2 µs | 11,0 µs |

Verwenden Sie den ACB-Befehl, um die Anzahl der Zeichen im Puffer zu bestimmen. Beim Übergang von unwahr nach wahr bestimmt die Steuerung die Gesamtzahl der Zeichen und fügt sie im POS-Feld des Steuerdatenfiles ein. Der Kanal muss für ASCII konfiguriert sein.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Channel (Kanal)** ist die Nummer des RS-232-Anschlusses, Kanal 0. (*Nur 1764-LRP: Auswahl von Kanal 0 oder Kanal 1 möglich.*).
- **Control (Steuerung)** ist der Steuerdatenfile. Siehe Seite 20-6.
- **Characters (Zeichen)** ist die Anzahl der von der Steuerung erkannten Zeichen im Puffer (0 bis 1024). Dieser Parameter ist schreibgeschützt.
- **Error (Fehler)** zeigt den hexadezimalen Fehlercode an, der angibt, warum das ER-Bit im Datenfile gesetzt wurde. Eine Beschreibung der Fehler finden Sie auf Seite 20-31.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.14 Gültige Adressierungsmodi und Filetypen für ACB-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Kanal | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Steuerung | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |

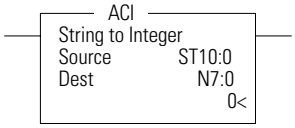
(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

## Funktionsweise

Wenn der Zustand des Strompfads von unwahr nach wahr übergeht, wird das EN-Bit gesetzt. Wenn der Befehl in die ASCII-Warteschlange gestellt wird, wird das Warteschlangen-Bit (EU) gesetzt. Das Ausfüh- rungs-Bit (RN) wird gesetzt, wenn der Befehl ausgeführt wird. Das Fertig-Bit (DN) wird gesetzt, wenn der Befehl abgeschlossen ist.

Die Steuerung bestimmt die Anzahl der Zeichen im Puffer und fügt diesen Wert in das POS-Feld des Steuerdatenfiles ein. Anschließend wird das Fertig-Bit (DN) gesetzt. Erscheint im POS-Feld eine Null, wurden keine Zeichen gefunden. Das Gefunden-Bit (FD) wird gesetzt, wenn das POS-Feld auf einen Wert ungleich Null gesetzt wurde.

# ACI – Zeichenkette in Ganzzahl




Befehlstyp: Ausgang

**Tabelle 20.15 Ausführungszeit des ACI-Befehls**

| Steuerung | Datengröße | Befehl | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | Wort | 17,6 µs + 7,2 µs/Zeichen | 0,0 µs |
| | Doppelwort | 24,6 µs + 11,6 µs/Zeichen | 0,0 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | | 14,2 µs + 6,3 µs/Zeichen | 0,0 µs |

Verwenden Sie den ACI-Befehl zum Konvertieren einer numerischen ASCII-Zeichenkette in einen ganzzahligen Wert (Wort oder Doppel- wort).

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Source (Quelle)** – Der Inhalt dieser Position wird in einen ganzzahligen Wert umgewandelt.
- **Destination (Zieladresse)** – An dieser Stelle wird das Ergebnis der Konvertierung eingefügt. Der Datenbereich liegt zwischen -32768 und 32767, wenn es sich bei der Zieladresse um ein Wort handelt. Ist die Zieladresse ein Doppelwort, liegt der Datenbereich zwischen -2 147 483 648 und 2 147 483 647.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.16 Gültige Adressierungsmodi und Filetypen für ACI-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Ziel | • | • | | • | • | | | | • | | | | | | | | | | | | | | | | • | | | • | • | |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

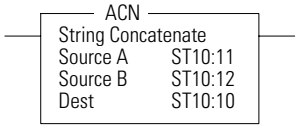
## Funktionsweise

Die Steuerung durchsucht die Quelle (Filetyp ST) nach dem ersten Zeichen, das zwischen 0 und 9 liegt. Alle numerischen Zeichen werden extrahiert, bis ein nicht numerisches Zeichen oder das Ende der Zeichenkette erkannt wird. Eine Aktion findet nur statt, wenn numerische Zeichen erkannt werden. Die Zeichenkettenlänge ist auf 82 Zeichen begrenzt. Kommata und Vorzeichen (+, –) sind nur in der Zeichenkette zulässig. Allerdings ist in der Tabelle nur das Minus- zeichen angeführt.

Dieser Befehl setzt die folgenden Mathematik-Flags im Steuerungsstatusfile:

| Mathematik-Flag | | Beschreibung |
|---|---|---|
| S:0/1 | Überlauf (V) | Flag wird gesetzt, wenn das Ergebnis außerhalb des gültigen Bereichs liegt. |
| S:0/2 | Null (Z) | Flag wird gesetzt, wenn das Ergebnis Null ist. |
| S:0/3 | Vorzeichen (S) | Flag wird gesetzt, wenn das Ergebnis negativ ist. |
| S:5/0 | Überlaufauffang | Das Flag wird gesetzt, wenn das Überlauf-Flag (S:0/1) gesetzt wird. |
| S:5/15 | Fehler-Bit für die ASCII-Zeichenket-tenbearbeitung | Das Flag wird gesetzt, wenn die Quellenzeichenkette mehr als 82 Zeichen enthält.<br>Ist S:5/15 gesetzt, wird der Fehler bezüglich der ungültigen Zeichenkettenlänge (1F39H) in den Haupt-Fehlercode (S:6) geschrieben. |

# ACN – Zeichenkette verketten




Befehlstyp: Ausgang

**Tabelle 20.17 Ausführungszeit des ACN-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | 22,6 µs + 11,5 µs/Zeichen | 0,0 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 17,9 µs + 10,2 µs/Zeichen | 0,0 µs |

Der ACN-Befehl verbindet zwei ASCII-Zeichenketten. Die zweite Zeichenkette wird an die erste Zeichenkette angehängt, und das Ergebnis an der Zieladresse gespeichert.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Source A (Quelle A**) ist die erste Zeichenkette im Verkettungsvorgang.
- **Source B (Quelle B**) ist die zweite Zeichenkette im Verkettungsvorgang.
- **Destination (Ziel)** ist die Position, an der das Ergebnis aus Quelle A und B gespeichert wird.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.18 Gültige Adressierungsmodi und Filetypen für ACN-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

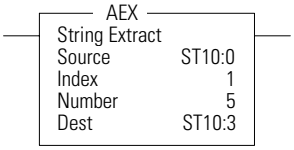
| Parameter | Datenfiles(1) | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Source (Quelle) A | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Source (Quelle) B | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Ziel | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

## Funktionsweise

Dieser Befehl wird beim Übergang von unwahr nach wahr ausgeführt. Quelle B wird an Quelle A angehängt, und das Ergebnis wird an die Zieladresse geschrieben. Es werden nur die ersten 82 Zeichen (0 bis 81) an die Zielposition geschrieben. Ist die Zeichenkettenlänge von Quelle A, Quelle B oder der Zieladresse größer als 82, wird das Fehler-Bit für die ASCII-Zeichenkettenbearbeitung (S:5/15) gesetzt und der Fehler hinsichtlich einer ungültigen Zeichenkettenlänge (1F39H) in das Haupt-Fehlercodewort (S:6) geschrieben.

# AEX – Zeichenkette extrahieren




Befehlstyp: Ausgang

**Tabelle 20.19 Ausführungszeit des AEX-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | 14,8 µs + 2,9 µs/Zeichen | 0,0 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 12,4 µs + 2,6 µs/Zeichen | 0,0 µs |

Der AEX-Befehl erzeugt eine neue Zeichenkette aus einem Teil einer vorhandenen Zeichenkette, der in einer neuen Zeichenkette gespeichert wird.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Source (Quelle)** ist die vorhandene Zeichenkette. Der Quellenwert wird durch diesen Befehl nicht beeinflusst.
- **Index** ist die Ausgangsposition (von 1 bis 82) der zu extra- hierenden Zeichenkette. (Ein Index von 1 weist auf das Zeichen ganz links in der Zeichenkette hin.)
- **Number (Anzahl)** ist die Anzahl der zu extrahierenden Zeichen (von 1 bis 82), beginnend bei der indizierten Position. Ist der Index zuzüglich der Anzahl größer als die Gesamtanzahl der Zeichen in der Quellenzeichenkette, entspricht die Zeichenkette an der Zieladresse den Zeichen vom Index bis zum Ende der Quellenzeichenkette.
- **Destination (Zieladresse)** ist das Zeichenkettenelement (ST), bei dem die extrahierte Zeichenkette gespeichert werden soll.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.20 Gültige Adressierungsmodi und Filetypen für AEX-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Index | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Nummer | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Ziel | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

## Funktionsweise

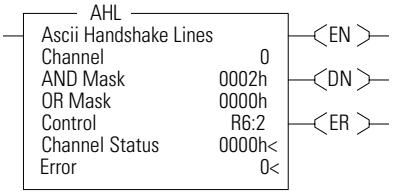
Dieser Befehl wird auf einem wahren Strompfad ausgeführt.

Die folgenden Bedingungen veranlassen die Steuerung, das Fehler-Bit für die ASCII-Zeichenkettenbearbeitung (S:5/15) zu setzen:

- Die Zeichenkettenlänge der Quelle ist kleiner als 1 oder größer als 82
- Der Wert für den Index ist kleiner als 1 oder größer als 82
- Der Wert für die Anzahl ist kleiner als 1 oder größer als 82
- Der Wert für den Index ist größer als die Länge der Quellenzeichenkette

Die Zielzeichenkette wird bei keiner der oben genannten Fehler- bedingungen geändert. Wenn das Fehler-Bit für die ASCII-Zeichenkettenbearbeitung (S:5/15) gesetzt ist, wird der Fehler bezüglich der ungültigen Zeichenkettenlänge (1F39H) in das Haupt-Fehlercodewort (S:6) geschrieben.

# AHL – ASCII-Handshake-Leitungen




Befehlstyp: Ausgang

**Tabelle 20.21 Ausführungszeit des AHL-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | 109,4 µs | 11,9 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 89,3 µs | 10,8 µs |

Der AHL-Befehl dient zum Setzen oder Zurücksetzen der Handshake-Steuerungsleitungen für die RS-232-Sendeaufforderung (RTS) eines Modems. Die Steuerung verwendet zwei Masken zur Bestimmung, ob die RTS-Steuerungszeile gesetzt bzw. zurückgesetzt werden soll oder unverändert bleibt. Der Kanal muss für ASCII konfiguriert sein.

**HINWEIS** Vergewissern Sie sich, dass die vom Anschluss verwendete automatische Modemsteuerung nicht zu Konflikten mit diesem Befehl führt.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Channel (Kanal)** ist die Nummer des RS-232-Anschlusses, Kanal 0. (*Nur 1764-LRP: Auswahl von Kanal 0 oder Kanal 1 möglich.*).

- **AND Mask (UND-Maske)** ist die Maske, die zum *Zurücksetzen* der RTS-Steuerungszeile verwendet wird. Bit 1 entspricht der RTS-Steuerungszeile. Der Wert „2“ in der UND-Maske setzt die RTS-Steuerungszeile zurück; mit dem Wert „0“ bleibt die Zeile unverändert.
- **OR Mask (ODER-Maske)** ist die Maske, die zum *Setzen* der RTS-Steuerungszeile dient. Bit 1 entspricht der RTS-Steuerungs- zeile. Der Wert „2“ in der ODER-Maske setzt die RTS-Steuerungs- zeile; mit dem Wert „0“ bleibt die Zeile unverändert.
- **Control (Steuerung)** ist der Steuerdatenfile. Siehe Seite 20-6.
- **Channel Status (Kanalstatus)** zeigt den aktuellen Status (0000 bis 001F) der Handshake-Leitungen für den angegebenen Kanal an. Dieser Status ist schreibgeschützt und wird in das .POS-Feld im Steuerdatenfile geschrieben. Die folgende Tabelle zeigt, wie der Wert des Kanalstatus bestimmt wird. In diesem Beispiel lautet der Wert 001F.

| **Kanalstatus-Bit** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Einstellung der Handshake-Steuerleitung | reserviert | | | | | | | | | | | -- | DCD(1) | -- | RTS | CTS |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| Kanalstatus | 0 | | | | 0 | | | | 1 | | | | F | | | |
| | Wort 2 des Steuerungselements = 001F | | | | | | | | | | | | | | | |

(1) Die DCD-Handshake-Leitung wird nur auf Kanal 1 unterstützt.

- **Error (Fehler)** zeigt den hexadezimalen Fehlercode an, der angibt, warum das ER-Bit im Datenfile gesetzt wurde. Eine Beschreibung der Fehlercodes finden Sie auf Seite 20-31.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.22 Gültige Adressierungsmodi und Filetypen für AHL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden“ auf Seite 4-2.*

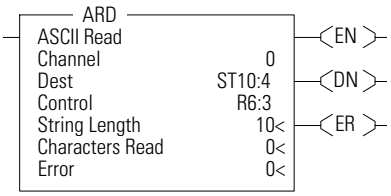
| **Parameter** | **Datenfiles(1)** | | | | | | | | | | | **Funktionsfiles** | | | | | | | | | **CS - Komm** | **IOS - E/A** | **DLS - Datenprotokoll** | **Adressierungsmodus** | | | **Adressierungsebene** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **O** | **I** | **S** | **B** | **T, C, R** | **N** | **F** | **ST** | **L** | **MG, PD** | **PLS** | **RTC** | **HSC** | **PTO, PWM** | **STI** | **EII** | **BHI** | **MMI** | **DAT** | **TPI** | | | | **unmittelbar** | **direkt** | **indirekt** | **Bit** | **Wort** | **Doppelwort** | **Element** |
| Kanal | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| UND-Maske | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | • | | | • | | |
| ODER-Maske | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | • | | | • | | |
| Steuerung | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

## Funktionsweise

Dieser Befehl wird auf wahren und unwahren Strompfaden ausge- führt. Für das Setzen des EN-Bits zum Wiederholen des Befehls ist jedoch ein Übergang des Strompfads von unwahr nach wahr erforderlich.

# ARD – ASCII-Lesen




Befehlstyp: Ausgang

**Tabelle 20.23 Ausführungszeit des ARD-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | 132,3 µs + 49,7 µs/Zeichen | 11,8 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 108 µs + 44 µs/Zeichen | 10,7 µs |

Verwenden Sie den ARD-Befehl zum Lesen von Zeichen aus dem Puffer und zum Speichern dieser Zeichen in einer Zeichenkette. Zum Wiederholen der Operation ist ein Übergang des Strompfades von unwahr nach wahr erforderlich.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Channel (Kanal)** ist die Nummer des RS-232-Anschlusses, Kanal 0. (*Nur 1764-LRP: Auswahl von Kanal 0 oder Kanal 1 möglich.*).
- **Destination (Zieladresse)** ist das Zeichenkettenelement, bei dem die Zeichen gespeichert werden sollen.
- **Control (Steuerung)** ist der Steuerdatenfile. Siehe Seite 20-6.
- **String Length (Zeichenkettenlänge)** (LEN) ist die Anzahl der Zeichen, die aus dem Puffer gelesen werden sollen. Die maximale Anzahl der Zeichen liegt bei 82. Wenn Sie eine Länge größer als 82 angeben, werden nur die ersten 82 Zeichen gelesen. Wenn Sie 0 Zeichen angeben, hat LEN standardmäßig den Wert 82. Dies ist Wort 1 in dem Steuerdatenfile.
- **Characters Read (Gelesene Zeichen)** (POS) ist die Anzahl der Zeichen, die die Steuerung aus dem Puffer in die Zeichenkette verschoben hat (0 bis 82). Dieses Feld wird während der Ausfüh- rung des Befehls aktualisiert und ist schreibgeschützt. Dies ist Wort 2 in dem Steuerdatenfile.
- **Error (Fehler)** zeigt den hexadezimalen Fehlercode an, der angibt, warum das ER-Bit im Datenfile gesetzt wurde. Eine Beschreibung der Fehlercodes finden Sie auf Seite 20-31.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.24 Gültige Adressierungsmodi und Filetypen für ARD-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Kanal | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Ziel | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Steuerung | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

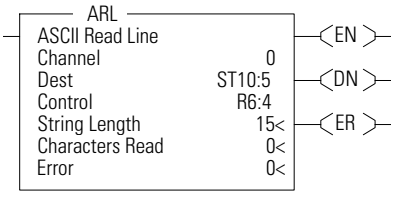
## Funktionsweise

Wenn der Zustand des Strompfads von unwahr nach wahr übergeht, wird das EN-Bit gesetzt. Wenn der Befehl in die ASCII-Warteschlange gestellt wird, wird das Warteschlangen-Bit (EU) gesetzt. Das Ausfüh- rungs-Bit (RN) wird gesetzt, wenn der Befehl ausgeführt wird. Das DN-Bit wird gesetzt, wenn der Befehl abgeschlossen ist.

Sobald sich die erforderliche Anzahl von Zeichen im Puffer befindet, werden die Zeichen an die Zielzeichenkette verschoben. Die Anzahl der verschobenen Zeichen wird in das POS-Feld des Steuerdatenfiles geschrieben. Die Zahl im POS-Feld wird ständig aktualisiert. Das Fertig-Bit (DN) wird erst gesetzt, wenn alle Zeichen gelesen wurden.

**HINWEIS** Informationen zur zeitlichen Abstimmung dieses Befehls finden Sie im Zeitdiagramm auf Seite 20-29.

# ARL – ASCII-Zeile lesen




Befehlstyp: Ausgang

**Tabelle 20.25 Ausführungszeit des ARL-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | 139,7 µs + 50,1 µs/Zeichen | 11,7 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 114 µs + 44,3 µs/Zeichen | 10,6 µs |

Verwenden Sie den ARL-Befehl zum Lesen der Zeichen aus dem Puffer (bis einschließlich der Abschlusszeichen) und zum Speichern der Zeichen in einer Zeichenkette. Die Abschlusszeichen werden über den Konfigurationsbildschirm des Kanals angegeben.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Channel (Kanal)** ist die Nummer des RS-232-Anschlusses, Kanal 0. (*Nur 1764-LRP: Auswahl von Kanal 0 oder Kanal 1 möglich.*).
- **Destination (Zieladresse)** ist das Zeichenkettenelement, bei dem die Zeichenkette gespeichert werden soll.
- **Control (Steuerung)** ist der Steuerdatenfile. Siehe Seite 20-6.
- **String Length (Zeichenkettenlänge)** (LEN) ist die Anzahl der Zeichen, die aus dem Puffer gelesen werden sollen. Die maximale Anzahl der Zeichen liegt bei 82. Wenn Sie eine Länge größer als 82 angeben, werden nur die ersten 82 Zeichen gelesen und an die Zieladresse verschoben. (Eine Länge von „0“ führt standardmäßig zu einem Wert von 82.) Dies ist Wort 1 in dem Steuerdatenfile.
- **Characters Read (Gelesene Zeichen)** (POS) ist die Anzahl der Zeichen, die die Steuerung aus dem Puffer in die Zeichenkette verschoben hat (0 bis 82). Dieses Feld ist schreibgeschützt und befindet sich in Wort 2 des Steuerdatenfiles.
- **Error (Fehler)** zeigt den hexadezimalen Fehlercode an, der angibt, warum das ER-Bit im Datenfile gesetzt wurde. Eine Beschreibung der Fehlercodes finden Sie auf Seite 20-31.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.26 Gültige Adressierungsmodi und Filetypen für ARL-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden“ auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Kanal | | | | | | | | | | | | | | | | | | | | | | | | • | | | | • | | |
| Ziel | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Steuerung | | | | | • | | | | | | | | | | | | | | | | | | | | • | | | | | • |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

## Funktionsweise

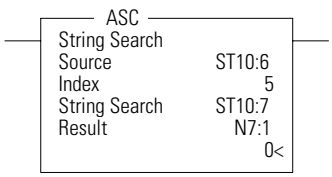
Bei einem Übergang des Strompfads von unwahr nach wahr wird das Freigabe-Bit (EN) des Steuerungselements gesetzt. Wenn der Befehl in die ASCII-Warteschlange gestellt wird, wird das Warteschlangen-Bit (EU) gesetzt. Das Ausführungs-Bit (RN) wird gesetzt, wenn der Befehl ausgeführt wird. Das DN-Bit wird gesetzt, wenn der Befehl abge- schlossen ist.

Sobald sich die erforderliche Anzahl von Zeichen im Puffer befindet, werden alle Zeichen (einschließlich der Abschlusszeichen) an die Zielzeichenkette verschoben. Die Anzahl der verschobenen Zeichen wird im POS-Wort des Steuerdatenfiles gespeichert. Die Zahl im Feld der gelesenen Zeichen (Characters Read) wird ständig aktualisiert. Das Fertig-Bit (DN) wird erst

gesetzt, wenn alle Zeichen gelesen wurden. Ausnahme: Findet die Steuerung Abschlusszeichen, bevor das Lesen beendet ist, wird das Fertig-Bit (DN) gesetzt, und die Anzahl der erkannten Zeichen wird im POS-Wort des Steuerdatenfiles gespei- chert.

**HINWEIS** Informationen zur zeitlichen Abstimmung dieses Befehls finden Sie im Zeitdiagramm auf Seite 20-29.

# ASC – Zeichenkette suchen




Befehlstyp: Ausgang

**Tabelle 20.27 Ausführungszeit des ASC-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | 16,2 µs + 4,0 µs/übereinstimmende Zeichen | 0,0 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 13,4 µs + 3,5 µs/übereinstimmende Zeichen | 0,0 µs |

Verwenden Sie den ASC-Befehl zum Durchsuchen einer vorhandenen Zeichenkette nach einem Vorkommen in der Quellenzeichenkette. Dieser Befehl wird auf einem wahren Strompfad ausgeführt.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Source (Quelle)** ist die Adresse der zu suchenden Zeichenkette.
- **Index** ist die Ausgangsposition (von 1 bis 82) innerhalb der Suchzeichenkette. (Ein Index von 1 weist auf das Zeichen ganz links in der Zeichenkette hin.)
- **Search (Suchen)** ist die Adresse der zu durchsuchenden Zeichenkette.
- **Result (Ergebnis)** ist die Position (von 1 bis 82), die die Steue- rung verwendet, um die Position in der Suchzeichenkette zu speichern, an der die Quellenzeichenkette beginnt. Kann keine Übereinstimmung gefunden werden, wird das Ergebnis gleich Null gesetzt.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

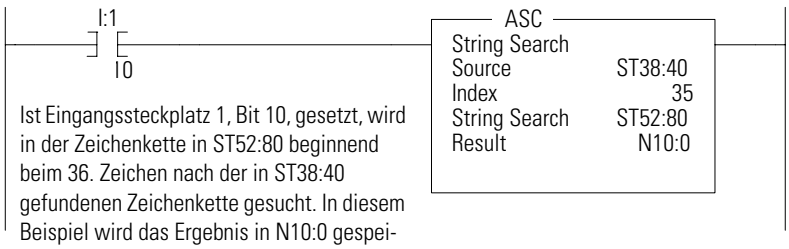
**Tabelle 20.28 Gültige Adressierungsmodi und Filetypen für ASC-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Quelle | • | • | | • | • | • | | | | | | | | | | | | | | | | | | • | • | | | • | | |
| Index | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Suchen | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Ergebnis | • | • | | • | • | • | | | | | | | | | | | | | | | | | | | • | | | • | | |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

## Beispiel:




Ist Eingangssteckplatz 1, Bit 10, gesetzt, wird in der Zeichenkette in ST52:80 beginnend beim 36. Zeichen nach der in ST38:40 gefundenen Zeichenkette gesucht. In diesem Beispiel wird das Ergebnis in N10:0 gespeichert.

## Fehlerzustände:

Die folgenden Bedingungen veranlassen die Steuerung, das ASCII-Fehler-Bit (S:5/15) zu setzen.

- Die Zeichenkettenlänge der Quelle ist kleiner als 1 oder größer als 82
- Der Wert für den Index ist kleiner als 1 oder größer als 82
- Der Wert für den Index ist größer als die Länge der Quellenzeichenkette

Das Ziel wird bei keinem der oben genannten Fehler geändert. Wenn das Fehler-Bit für die ASCII-Zeichenkettenbearbeitung (S:5/15) gesetzt ist, wird der Fehler bezüglich der ungültigen Zeichenkettenlänge (1F39H) in das Haupt-Fehlercodewort (S:6) geschrieben.

# ASR – ASCII-Zeichenkette vergleichen

Befehlstyp: Eingang

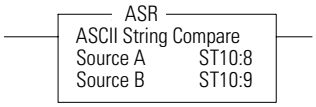


**Tabelle 20.29 Ausführungszeit des ASR-Befehls**

| Steuerung | Befehl | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 Serie B, FRN 3 oder höher | 9,2 µs + 4,0 µs/übereinstimmende Zeichen | 0,0 µs |
| MicroLogix 1500 Serie B, FRN 4 oder höher | 7,5 µs + 3,5 µs/übereinstimmende Zeichen | 0,0 µs |

Verwenden Sie den ASR-Befehl zum Vergleichen zweier ASCII-Zeichenketten. Die Steuerung sucht nach einer übereinstimmenden Länge und übereinstimmenden Groß- bzw. Kleinbuchstaben. Sind zwei Zeichenketten identisch, ist der Strompfad wahr. Liegen Unterschiede vor, ist der Strompfad unwahr.

## Parameter eingeben

Geben Sie beim Programmieren dieses Befehls die folgenden Parameter ein:

- **Source A (Quelle A)** ist die Position der ersten Zeichenkette, die für den Vergleich verwendet wird.
- **Source B (Quelle B)** ist die Position der zweiten Zeichenkette, die für den Vergleich verwendet wird.

Adressierungsmodi und Filetypen können entsprechend der folgenden Tabelle verwendet werden:

**Tabelle 20.30 Gültige Adressierungsmodi und Filetypen für ASR-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden" auf Seite 4-2.*

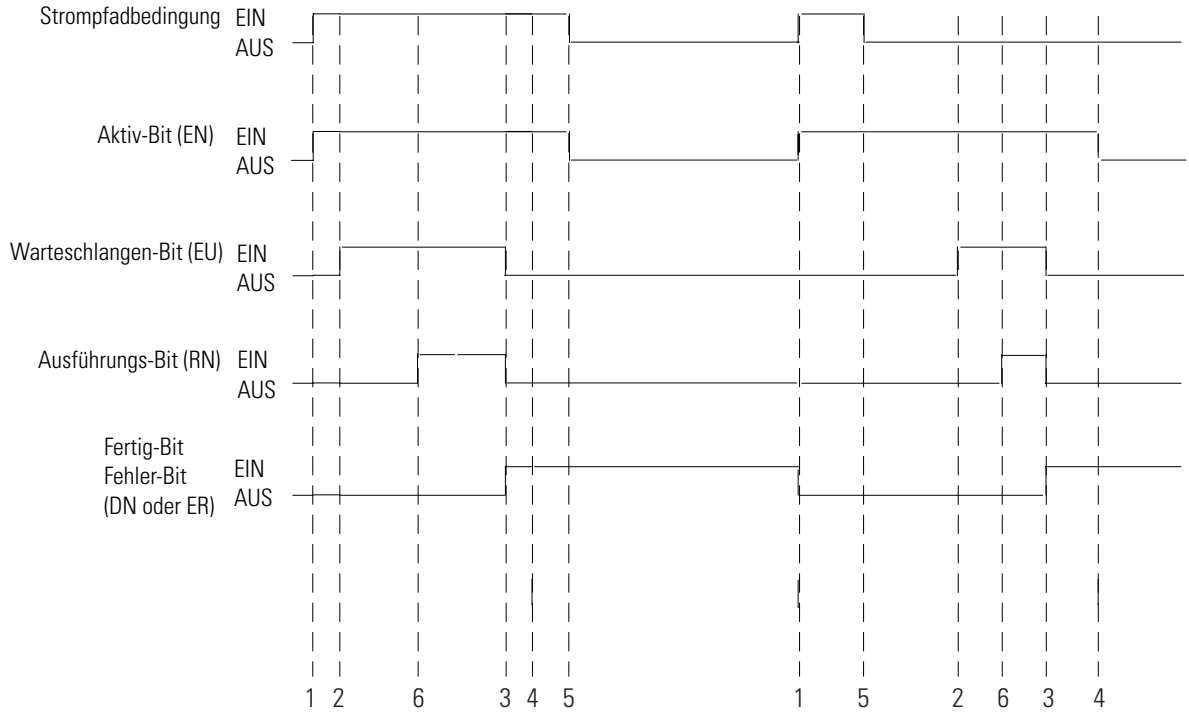
| Parameter | Datenfiles[1] | | | | | | | | | | | Funktionsfiles | | | | | | | | | CS - Komm | IOS - E/A | DLS - Datenprotokoll | Adressierungsmodus | | | Adressierungsebene | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | PLS | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | | | | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Element |
| Source (Quelle) A | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |
| Source (Quelle) B | | | | | | | | • | | | | | | | | | | | | | | | | | • | | | | | • |

(1) Der Steuerdatenfile ist der einzig gültige Filetyp für das Steuerungselement.

## Funktionsweise

Ist die Zeichenkette von Quelle A oder Quelle B länger als 82 Zeichen, wird das Fehler-Bit für die ASCII-Zeichenkettenbearbeitung (S:5/15) gesetzt, und der Zustand des Strompfads ändert sich in unwahr.

# Zeitdiagramm für die ARD-, ARL-, AWA- und AWT-Befehle




HINWEIS: Das Ausführungs-Bit (RN) kann nicht über den Control (R)-File adressiert werden.

1 - Strompfad wird wahr
2 - Befehl erfolgreich in die Warteschlange gestellt
3 - Befehlsausführung abgeschlossen
4 - Befehl nach Abschluss der Ausführung zum ersten Mal abgefragt
5 - Strompfad wird unwahr
6 - Befehlsausführung beginnt

## Verwenden der eingeschleiften indirekten Adressierung

Hierbei können Sie Ganzzahlwerte und Doppelwortwerte in ASCII-Zeichenketten einfügen. Der Zeichenkettenwert kann nur verwendet werden, wenn das Ausführungs-Bit (RN) gesetzt ist.

Für die eingeschleifte indirekte Adressierung gelten folgende Regeln:

- Alle gültigen Ganzzahl- und Doppelwortfiles (N und L) können verwendet werden.
  Gültigkeitsbereich: 3 bis 255.
- Bei den Filetypen wird keine Unterscheidung nach Groß- oder Kleinschreibung gemacht; dabei kann ein Doppelpunkt (:) oder ein Semikolon (;) verwendet werden.
- Das Pluszeichen (+) und führende Nullen werden nicht gedruckt. Negative Wert (-) werden mit einem vorangestellten Minuszeichen gedruckt. In Zahlen, die größer sind als 1000, werden keine Tausendertrennzeichen eingefügt.

*Beispiele:*

Für die nachfolgenden Beispiele gilt:
N7:0 = 25
N7:1 = -37
L8:0 = 508000
L8:1 = 5

| **Gültige eingeschleifte indirekte Adressierung:** | |
|---|---|
| Eingang: | Die aktuelle Strömungsrate beträgt [N7:0] Liter pro Minute mit einem Verschmutzungsgrad von [L8:0] Partikeln pro Liter. |
| Ausgang: | Die aktuelle Strömungsrate beträgt 25 Liter pro Minute mit einem Verschmutzungsgrad von 508000 Partikeln pro Liter. |
| Eingang: | Die aktuelle Position ist [N7:1] bei einer Geschwindigkeit von [L8:1] U/min. |
| Ausgang: | Die aktuelle Position ist -37 bei einer Geschwindigkeit von 5 U/min. |
| **Ungültige eingeschleifte indirekte Adressierung:** | |
| Eingang: | Die aktuelle Position ist [N7:1] bei einer Geschwindigkeit von [L8:1] U/min. |
| Ausgang: | Die aktuelle Position ist [N5:1] bei einer Geschwindigkeit von 5 U/min. |

**HINWEIS** Die Ausgangszeichenkette wird gekürzt, wenn auf- grund der indirekten Adressierung ein Ausgang mit mehr als 82 Zeichen entsteht. Die angefügten Zeichen werden immer für den Ausgang angewandt.

## Fehlercodes zu ASCII-Befehlen

Folgende Fehlercodes zeigen an, weshalb das Fehler-Bit (ER) in dem Steuerdatenfile gesetzt wurde.

| Fehlercode | | Beschreibung | Empfohlene Abhilfemaßnahme |
|---|---|---|---|
| **dezimal** | **hexadezimal** | | |
| 0 | 0x00 | Kein Fehler. Der Befehl wurde erfolgreich abgeschlossen. | Nicht erforderlich. |
| 3 | 0x03 | Die Übertragung kann nicht abgeschlossen werden, da das CTS-Signal nicht mehr vorhanden ist. | Überprüfen Sie das Modem und die Modemverbindungen. |
| 5 | 0x05 | Bei der Durchführung einer ASCII-Übertragung wurde ein Konflikt mit dem konfigurierten Kommunikationsprotokoll festgestellt. | Konfigurieren Sie den Kanal erneut, und wiederholen Sie die Operation. |
| 7 | 0x07 | Der Befehl kann nicht ausgeführt werden, da der Kommunikationskanal durch das Konfigurationsmenü für den Kanal geschlossen wurde. | Konfigurieren Sie den Kanal erneut, und wiederholen Sie die Operation. |
| 8 | 0x08 | Der Befehl kann nicht ausgeführt werden, da bereits eine andere ASCII-Übertragung läuft. | Starten Sie die Übertragung erneut. |
| 9 | 0x09 | Der angeforderte Typ der ASCII-Kommunikationsfunktion wird durch die aktuelle Kanalkonfiguration nicht unterstützt. | Konfigurieren Sie den Kanal erneut, und wiederholen Sie die Operation. |
| 10 | 0x0A | Das Entlade-Bit (UL) wurde gesetzt und die Ausführung des Befehls gestoppt. | Nicht erforderlich. |
| 11 | 0x0B | Die angeforderte Anzahl von Zeichen war für die gelesene ASCII-Zeichenkette zu groß oder negativ. | Geben Sie eine gültige Zeichenkettenlänge ein, und geben Sie den Befehl erneut ein. |
| 12 | 0x0C | Die Länge der Quellenzeichenkette ist ungültig (negativ oder größer als 82). | Geben Sie eine gültige Zeichenkettenlänge ein, und geben Sie den Befehl erneut ein. |
| 13 | 0x0D | Die angeforderte Länge im Steuerungsfeld ist ungültig (entweder ein negativer Wert oder eine Zahl größer als 82). | Geben Sie eine gültige Länge ein, und geben Sie den Befehl erneut ein. |
| 14 | 0x0E | Die Ausführung eines ACL-Befehls führte zum Abbruch dieses Befehls. | Nicht erforderlich. |
| 15 | 0x0F | Die Konfiguration des Kommunikationskanals wurde während der Ausführung des Befehls geändert. | Nicht erforderlich. |

## ASCII-Zeichensatz

In der folgenden Tabelle sind die Dezimal-, Hexadezimal- und Oktalwerte und deren ASCII-Umwandlungswerte angegeben.

**Tabelle 20.31 ASCII-Standardzeichensatz**

| **Spalte 1** | | | | | **Spalte 2** | | | | **Spalte 3** | | | | **Spalte 4** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Strg- | DEZ | HEX | OKT | ASC | DEZ | HEX | OKT | ASC | DEZ | HEX | OKT | ASC | DEZ | HEX | OKT | ASC |
| ^@ | 00 | 00 | 000 | NUL | 32 | 20 | 040 | SP | 64 | 40 | 100 | @ | 96 | 60 | 140 | \ |
| ^A | 01 | 01 | 001 | SOH | 33 | 21 | 041 | ! | 65 | 41 | 101 | A | 97 | 61 | 141 | a |
| ^B | 02 | 02 | 002 | STX | 34 | 22 | 042 | „ | 66 | 42 | 102 | B | 98 | 62 | 142 | b |
| ^C | 03 | 03 | 003 | ETX | 35 | 23 | 043 | # | 67 | 43 | 103 | C | 99 | 63 | 143 | c |
| ^D | 04 | 04 | 004 | EOT | 36 | 24 | 044 | $ | 68 | 44 | 104 | D | 100 | 64 | 144 | d |
| ^E | 05 | 05 | 005 | ENQ | 37 | 25 | 045 | % | 69 | 45 | 105 | E | 101 | 65 | 145 | e |
| ^F | 06 | 06 | 006 | ACK | 38 | 26 | 046 | & | 70 | 46 | 106 | F | 102 | 66 | 146 | f |
| ^G | 07 | 07 | 007 | BEL | 39 | 27 | 047 | ' | 71 | 47 | 107 | G | 103 | 67 | 147 | g |
| ^H | 08 | 08 | 010 | BS | 40 | 28 | 050 | ( | 72 | 48 | 110 | H | 104 | 68 | 150 | h |
| ^I | 09 | 09 | 011 | HT | 41 | 29 | 051 | ) | 73 | 49 | 111 | I | 105 | 69 | 151 | i |
| ^J | 10 | 0A | 012 | LF | 42 | 2A | 052 | * | 74 | 4A | 112 | J | 106 | 6A | 152 | j |
| ^K | 11 | 0B | 013 | VT | 43 | 2B | 053 | + | 75 | 4B | 113 | K | 107 | 6B | 153 | k |
| ^L | 12 | 0C | 014 | FF | 44 | 2C | 054 | , | 76 | 4C | 114 | L | 108 | 6C | 154 | l |
| ^M | 13 | 0D | 015 | CR | 45 | 2D | 055 | - | 77 | 4D | 115 | M | 109 | 6D | 155 | m |
| ^N | 14 | 0E | 016 | SO | 46 | 2E | 056 | . | 78 | 4E | 116 | N | 110 | 6E | 156 | n |
| ^O | 15 | 0F | 017 | SI | 47 | 2F | 057 | / | 79 | 4F | 117 | O | 111 | 6F | 157 | o |
| ^P | 16 | 10 | 020 | DLE | 48 | 30 | 060 | 0 | 80 | 50 | 120 | P | 112 | 70 | 160 | p |
| ^Q | 17 | 11 | 021 | DC1 | 49 | 31 | 061 | 1 | 81 | 51 | 121 | Q | 113 | 71 | 161 | q |
| ^R | 18 | 12 | 022 | DC2 | 50 | 32 | 062 | 2 | 82 | 52 | 122 | R | 114 | 72 | 162 | r |
| ^S | 19 | 13 | 023 | DC3 | 51 | 33 | 063 | 3 | 83 | 53 | 123 | S | 115 | 73 | 163 | s |
| ^T | 20 | 14 | 024 | DC4 | 52 | 34 | 064 | 4 | 84 | 54 | 124 | T | 116 | 74 | 164 | t |
| ^U | 21 | 15 | 025 | NAK | 53 | 35 | 065 | 5 | 85 | 55 | 125 | U | 117 | 75 | 165 | u |
| ^V | 22 | 16 | 026 | SYN | 54 | 36 | 066 | 6 | 86 | 56 | 126 | V | 118 | 76 | 166 | v |
| ^W | 23 | 17 | 027 | ETB | 55 | 37 | 067 | 7 | 87 | 57 | 127 | W | 119 | 77 | 167 | w |
| ^X | 24 | 18 | 030 | CAN | 56 | 38 | 070 | 8 | 88 | 58 | 130 | X | 120 | 78 | 170 | x |
| ^Y | 25 | 19 | 031 | EM | 57 | 39 | 071 | 9 | 89 | 59 | 131 | Y | 121 | 79 | 171 | y |
| ^Z | 26 | 1A | 032 | SUB | 58 | 3A | 072 | : | 90 | 5A | 132 | Z | 122 | 7A | 172 | z |
| ^[ | 27 | 1B | 033 | ESC | 59 | 3B | 073 | ; | 91 | 5B | 133 | [ | 123 | 7B | 173 | { |
| ^\ | 28 | 1C | 034 | FS | 60 | 3C | 074 | < | 92 | 5C | 134 | \ | 124 | 7C | 174 | \| |
| ^] | 29 | 1D | 035 | GS | 61 | 3D | 075 | = | 93 | 5D | 135 | ] | 125 | 7D | 175 | } |
| ^^ | 30 | 1E | 036 | RS | 62 | 3E | 076 | > | 94 | 5E | 136 | ^ | 126 | 7E | 176 | ~ |
| ^_ | 31 | 1F | 037 | US | 63 | 3F | 077 | ? | 95 | 5F | 137 | _ | 127 | 7F | 177 | DEL |

Der ASCII-Standardzeichensatz enthält Werte bis zu 127 dezimal (7F hex). Die MicroLogix 1200- und 1500-Steuerungen unterstützen auch einen erweiterten Zeichensatz (Dezimalwert 128 bis 255). In Abhängigkeit von der verwendeten Plattform können jedoch bei dem erweiterten Zeichensatz andere Zeichen angezeigt werden.

Den Dezimalwerten 0 bis 31 kann auch ein Strg-Code zugewiesen werden.

# Kommunikationsbefehle

Dieses Kapitel enthält Informationen über die Kommunikations- befehle MSG (Nachricht) und SVC (Kommunikationsbearbeitung). Dabei werden folgende Themen behandelt:


- „Nachrichtenfunktion – Übersicht“ auf Seite 21-1
- „SVC – Kommunikationsbearbeitung“ auf Seite 21-3
- „MSG – Nachricht“ auf Seite 21-5
- „Nachrichten-Element“ auf Seite 21-6
- „Zeitdiagramm für MSG-Befehl“ auf Seite 21-12
- „Kontaktplanlogik für MSG-Befehl“ auf Seite 21-15
- „Zentrale Nachrichten“ auf Seite 21-16
- „Zentrale Nachrichtenübertragung konfigurieren“ auf Seite 21-18
- „Beispiele für zentrale Nachrichten“ auf Seite 21-25
- „Dezentrale Nachrichten“ auf Seite 21-37
- „Dezentrale Nachrichtenübertragung konfigurieren“ auf Seite 21-39
- „Fehlercodes zu MSG-Befehlen“ auf Seite 21-43


Die Kommunikationsbefehle lesen Daten aus einer und schreiben Daten in eine andere Station.

| Befehl | Zweck | Seite |
|---|---|---|
| SVC | Programmabfrage für die Bearbeitung der Kommunikation innerhalb des Betriebszyklus unterbrechen. Anschließend wird die Programmabfrage bei dem Befehl fortgesetzt, der dem SVC-Befehl unmittelbar folgt. | 21-3 |
| MSG | Übertragung von Daten von einem Gerät zu einem anderen. | 21-5 |

## Nachrichtenfunktion – Übersicht

Die Kommunikationsarchitektur besteht aus drei Hauptkomponenten:

- Kontaktplanabfrage
- Kommunikationspuffer
- Kommunikationswarteschlange

Diese drei Komponenten bestimmen, wann eine Nachricht durch die Steuerung übertragen wird. Eine Nachricht wird nur übertragen, wenn sie auf einem wahren Kontaktplanstrompfad abgefragt wurde. Bei der Abfrage werden die Nachricht und die in der Nachricht enthaltenen Daten (sofern es sich um eine Schreibnachricht handelt) in einen Kommunikationspuffer gestellt. Die Steuerung setzt die Abfrage des übrigen Anwenderprogramms fort. Die Nachricht wird verarbeitet und über den Kommunikationsanschluss nach Abschluss der Kontakt- planlogik, während des Kommunikationsteils des

Funktionszyklus und bis zum Ausführen eines SVC-Befehls an die Steuerung gesendet.

Wenn vor Abschluss der ersten Nachricht eine zweite Nachricht verarbeitet wird, werden die zweite Nachricht und die darin enthal- tenen Daten in einen der drei anderen Kommunikationspuffer gestellt. Dieser Vorgang wiederholt sich, sobald ein Nachrichtenbefehl verarbeitet wird, bis alle vier Puffer belegt sind.

Wenn ein Puffer verfügbar ist, werden die Nachricht und die zuge- hörigen Daten sofort in den Puffer gestellt. Sind alle vier Puffer des Kanals voll, wenn die nächste (fünfte) Nachricht verarbeitet wird, stellt das System die Nachrichtenaufforderung (nicht die Daten) in die Kommunikationswarteschlange des Kanals. Die Warteschlange ist ein Speicherbereich für Nachrichten, die keinem Puffer zugeordnet wurden. Für die Warteschlange gilt das FIFO-Prinzip (First-In First-Out). Die erste Nachrichtenübertragungsaufforderung, die in die Warteschlange gestellt wurde, wird dem nächsten frei werdenden Puffer zugeordnet. Alle MSG-Befehle in einem Kontaktplanprogramm können in die Warteschlange aufgenommen werden.

Nach Abschluss einer Nachrichtenübertragungsaufforderung in einem Puffer wird der Puffer wieder dem System zur Verfügung gestellt. Wenn sich eine Nachricht in der Warteschlange befindet, wird die Nachricht dem Puffer zugewiesen. Zu diesem Zeitpunkt werden auch die zu der Nachricht gehörenden Daten aus der Steuerung eingelesen.

**HINWEIS** Wenn sich ein Nachrichtenbefehl in der Warteschlange befand, können sich die von der Steuerung verschick- ten Daten von denen unterscheiden, die bei der ursprünglichen Verarbeitung des Befehls vorlagen.

Der Puffer und die Warteschlange arbeiten vollautomatisch. Puffer werden entsprechend des jeweiligen Bedarfs zugeordnet und freigegeben; sobald alle Puffer belegt sind, werden die weiteren Nachrichten in die Warteschlange gestellt.

Die Steuerung initiiert Lese- und Schreibnachrichten über verfügbare Kommunikationskanäle, wenn sie für die folgenden Protokolle konfiguriert wurde:

- DH-485
- DF1-Vollduplex
- DF1 Halbduplex Slave

Eine Beschreibung der gültigen Kommunikationsprotokolle finden Sie unter „Protokollkonfiguration“ auf Seite E-1.

# SVC – Kommunikations-bearbeitung

Befehlstyp: Ausgang

**Tabelle 21.1 Ausführungszeit des SVC-Befehls**

| Steuerung | Strompfad:(1) | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1200 | 208 µs + 1,6 µs pro Wort | 0,0 µs |
| MicroLogix 1500 1764-LSP oder 1764-LRP (ein Kanal ausgewählt) | 166 µs + 1,4 µs pro Wort | 0,0 µs |
| MicroLogix 1500 1764-LRP-Prozessor (beide Kanäle ausgewählt) | 327 µs + 1,4 µs pro Wort | 0,0 µs |

(1) Dieser Wert für den Befehl SVC wird gesetzt, wenn der Kommunikationsdienst auf einen Datenfile zugreift. Die Zeit erhöht sich beim Zugreifen auf einen Funktionsfile.

Im Normalbetrieb wird die Kommunikation von der Steuerung bei jeder Abfrage des Steuerprogramms bearbeitet. Wenn die Kommunikationsanschlüsse öfter abgefragt werden sollen oder wenn die Kontaktplanabfrage sehr lange dauert, können Sie einen SVC-Befehl (Kommunikationsbearbeitung) in das Steuerprogramm aufnehmen. Der SVC-Befehl wird zur Verbesserung der Kommunikationsleistung (Durchsatz) verwendet, verlängert jedoch auch die Kontaktplan-abfrage.

Stellen Sie den SVC-Befehl einfach auf einen Strompfad innerhalb des Steuerprogramms. Bei der Abfrage des Strompfads wird jegliche Kommunikationsanforderung abgearbeitet. Dabei kann der SVC-Befehl auf einen Strompfad ohne vorhergehende Logik gestellt oder mit Hilfe verschiedener Kommunikationsstatus-Bits ein bedingter Strompfad eingerichtet werden. Die Tabelle auf Seite 21-4 enthält die verfügbaren Statusfile-Bits.

**HINWEIS** Der Umfang der Kommunikationsbearbeitung wird durch das CSS-Bit (Communication Servicing Selection) und das MSS-Bit (Message Servicing Selection) in dem Kommunikations-Konfigurationsfile für Kanal 0 bestimmt.

Die besten Ergebnisse erzielen Sie, wenn Sie den SVC-Befehl in der Mitte des Steuerungsprogramms platzieren. SVC-Befehle dürfen nicht in Fehler-, DII-, STI- oder E/A-Ereignissubroutinen eingefügt werden.

## Kanalauswahl

Bei Verwendung des SVC-Befehls muss der Kanal für die Kommunikationsbearbeitung ausgewählt werden. Die Kanalauswahlvariable ist ein Ein-Wort-Bitmuster, das den zu bearbeitenden Kanal bezeichnet. Jedes Bit entspricht einem bestimmten Kanal. Beispielsweise steht Bit 0 für Kanal 0. Wenn ein Bit gesetzt ist (1), wird der entsprechende Kanal bearbeitet.

| Steuerung | Einstellung für Kanalauswahl | Betriebene(r) Kanal/Kanäle |
|---|---|---|
| MicroLogix 1200 | 1 | 0 |
| MicroLogix 1500 mit 1764-LSP-Prozessor | 1 | 0 |
| MicroLogix 1500 mit 1764-LRP-Prozessor | 1 | 0 |
| | 2 | 1 |
| | 3 | 0 und 1 |

## Kommunikationsstatus-Bits

Mit den folgenden Kommunikationsstatus-Bits können Sie die Kommunikationsbearbeitung individuell anpassen oder überwachen. Weitere Statusinformationen finden Sie im Abschnitt „Block zum allgemeinen Kanalstatus“ auf Seite 3-15.

**Tabelle 21.2 Kommunikations-Status-Bits**

| Adresse | | Beschreibung |
|---|---|---|
| **Kanal 0** | **Kanal 1(1)** | |
| CS0:4/0 | CS1:4/0 | ICP - Incoming Command Pending (Eingehender Befehl anstehend) |
| CS0:4/1 | CS1:4/1 | MRP - Incoming Message Reply Pending (Eingehende Nachricht Antwort anstehend) |
| CS0:4/2 | CS1:4/2 | MCP - Outgoing Message Command Pending (Ausgehende Nachricht Befehl anstehend) |
| CS0:4/4 | CS1:4/4 | CAB - Communications Active Bit (Bit für aktive Kommunikation) |

(1) Kanal 1 ist nur für MicroLogix 1500 1764-LRP gültig.

## Anwendungsbeispiel

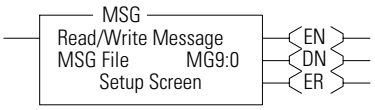
Verwenden Sie den SVC-Befehl, wenn Sie eine Kommunikations- funktion, wie beispielsweise die Übertragung einer Nachricht, vor dem normalen Kommunikations-Bearbeitungsteil des Abarbeitungszyklus ausführen möchten.

Sie können diesen Strompfad nach einem Nachricht-Schreibbefehl programmieren. CS0:4/MCP wird gesetzt, wenn der Nachrichtenbefehl aktiviert und in die Kommunikationswarteschlange gestellt wird. Wenn CS0:4/MCP gesetzt ist (1), wird der SVC-Befehl als wahr bewertet, und die Programmabfrage wird für die Ausführung der Kommunikationsabarbeitung im Rahmen der Abarbeitungsabfrage unterbrochen. Anschließend wird die Programmabfrage bei dem Befehl fortgesetzt, der dem SVC-Befehl unmittelbar folgt.

Der Beispielstrompfad zeigt einen bedingten SVC-Befehl, der nur verarbeitet wird, wenn sich eine ausgehende Nachricht in der Kommunikationswarteschlange befindet.

**HINWEIS** Sie können den SVC-Befehl auch in unbedingten Strompfaden programmieren. Dies ist das normale Programmierverfahren für SVC-Befehle.

# MSG – Nachricht




Befehlstyp: Ausgang

**Tabelle 21.3 Ausführungszeit des MSG-Befehls**

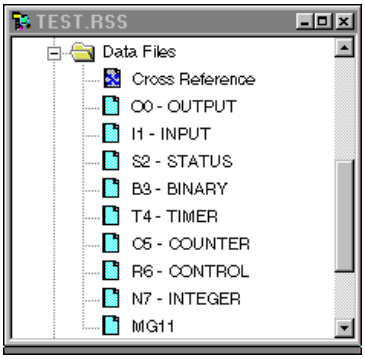
| Steuerung | Strompfadbedingung | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1200 | ständig wahr | 20,0 µs | 6,0 µs |
| | Strompfadübergang von unwahr nach wahr bei Lesenachrichten | 230,0 µs | |
| | Strompfadübergang von unwahr nach wahr bei Schreibnachrichten | 264 µs + 1,6 µs pro Wort | |
| MicroLogix 1500 1764-LSP | ständig wahr | 17,0 µs | 6,0 µs |
| | Strompfadübergang von unwahr nach wahr bei Lesenachrichten | 205,0 µs | |
| | Strompfadübergang von unwahr nach wahr bei Schreibnachrichten | 228 µs + 1,4 µs pro Wort | |
| MicroLogix 1500 1764-LRP | ständig wahr | 17,0 µs | 6,0 µs |
| | *Kommunikation über Basisgerät oder 1764-LRF Kommunikationsanschluss:* | | |
| | Strompfadübergang von unwahr nach wahr bei Lesenachrichten | 234,0 µs | 6,0 µs |
| | Strompfadübergang von unwahr nach wahr bei Schreibnachrichten | 257 µs + 1,4 µs pro Wort | |
| | *Kommunikation über Kompakt E/A-Kommunikationsmodul, d. h. 1769-SDN:* | | |
| | Strompfadübergang von unwahr nach wahr bei Lesenachrichten | 206,0 µs | 6,0 µs |
| | Strompfadübergang von unwahr nach wahr bei Schreibnachrichten | 234 µs + 1,4 µs pro Wort | |

Der Nachrichtenbefehl kann nur verarbeitet werden, wenn jegliche vorherige Logik auf dem Nachrichtenstrompfad wahr ist. Das nachfol- gende Beispiel zeigt einen Nachrichtenbefehl.

Wenn B3/0 aktiviert (1) ist, der MSG-Strompfad wahr ist und MG11:0 noch keine Nachricht verarbeitet, wird MG11:0 verarbeitet. Wenn einer der vier Puffer verfügbar ist, werden die Nachricht und die zugehörigen Daten sofort verarbeitet.

**HINWEIS** Die Geschwindigkeit der Übertragung der Nachricht an das Zielgerät wird von mehreren Faktoren bestimmt; hierzu gehören das für den Kanal ausgewählte Kommunikationsprotokoll, die Baudrate des Kommunikationsanschlusses, die Anzahl der gegebenenfalls erforderlichen Wiederholungen und die Fähigkeit des Zielgeräts, die Nachricht zu empfangen.

# Nachrichten-Element




Der in die Steuerung integrierte MSG-Befehl verwendet einen MG-Datenfile für die Verarbeitung des Nachrichtenbefehls. Der Zugriff auf den MG-Datenfile (siehe Abbildung links) erfolgt über die Kennung MG. Jeder Nachrichtenbefehl verwendet ein Element innerhalb eines MG-Datenfiles. Zum Beispiel ist MG11:0 das erste Element in dem Nachrichten-Datenfile 11.

## Unterelemente des Nachrichtenfiles

Jeder MSG-Befehl muss ein eindeutiges Element in einem MSG-File verwenden. Das MSG-Element jedes MSG-Befehls enthält alle Para- meter und Statusinformationen für diesen speziellen MSG-Befehl.

Jedes MSG-Fileelement besteht aus den Unterelementen 0 bis 24 (siehe folgende Tabelle).

**Element des Nachrichtenfiles**

| Unter-element | Name | Beschreibung | Parameter | Größe | Anwender-programm-zugriff[1] |
|---|---|---|---|---|---|
| 0 bis 1 | | Reserviert | | Wort | Nur Lesen |
| 2 | | Nachrichtentyp: 0 (für PCCC), 1 (für CIP) | | Wort | Nur Lesen |
| 3 | | für PCCC-Nachrichten: Bits 07–00 (CMD-Code), Bits 15–08 (FNC-Code)<br>für CIP-Nachrichten: Bits 07–00 (Servicecode), Bits 15–08 (zusätzliche Datenanzahl auf Objektpfad) | Abgeleitet | Wort | Nur Lesen |
| 4 | | Reserviert | | Wort | Nur Lesen |
| 5 | MG11:0.RBL | Verbundkennung der dezentralen Brücken/Zusätzliche Datenanzahl auf Objektpfad Bytes 0 und 1 | J | Wort | Nur Lesen |
| 6 | MG11:0.LBN | Netzknotenadressen der lokalen Brücken/Zusätzliche Datenanzahl auf Objektpfad Bytes 2 und 3 | J | Wort | Nur Lesen |
| 7 | MG11:0.RBN | Netzknotenadressen der dezentralen Brücken/Zusätzliche Datenanzahl auf Objektpfad Bytes 4 und 5 | J | Wort | Nur Lesen |
| 8 | MG11:0.CHN | Kanal: Bits 07–00 (0 für Kanal 0, 1 für Kanal 1)<br>Steckplatz: Bits 15–08 (0 bis 16) | J | Wort | Lesen/ Schreiben |
| 9 | MG11:0.NOD | Nummer des Zielnetzknotens | J | Wort | Lesen/ Schreiben |
| 10 | MG11:0.MTO | Einstellung für Nachrichten-Zeitablauf oder Voreinstellung in Sekunden | J | Wort | Lesen/ Schreiben |
| 11 | | Anzahl der zu lesenden/schreibenden Bytes | | Wort | Nur Lesen |
| 12 | | Daten zur Zielposition(Optionen enthalten die Tabellen auf Seite 21-7 ) | J | Wort | Nur Lesen |
| 13 | MG11:0.TFN | | J | Wort | Lesen/ Schreiben |
| 14 | MG11:0.ELE | | J | Wort | Lesen/ Schreiben |
| 15 | | | J | Wort | Nur Lesen |
| 16 | | Steuer-Bits (ausführliche Informationen enthält die Tabelle zu den Steuer-Bits auf Seite 21-8) | N | 16 Bits | Lesen/ Schreiben |
| 17 | | Status-Bits und Bereichsparameter (ausführliche Informationen enthält die Tabelle auf Seite 21-9) | Ver-schiedene | 16 Bits | Nur Lesen |
| 18 | MG11:0.ERR | Fehlercode (siehe „Fehlercodes“ auf Seite 21-43) | N | Wort | Nur Lesen |

| Element des Nachrichtenfiles | | | | | |
|---|---|---|---|---|---|
| **Unter-element** | **Name** | **Beschreibung** | **Parameter** | **Größe** | **Anwender-programm-zugriff[1]** |
| 19 | | Zeit seit Nachrichtenbeginn in Sekunden | N | Wort | Nur Lesen |
| 20 | | Reserviert | | Wort | Nur Lesen |
| 21 | | Interne Nachrichtenstartzeit in Sekunden | N | Wort | Nur Lesen |
| 22 | | Reserviert | N | Wort | Nur Lesen |
| 23 | | *Nur für MicroLogix 1500 1764-LRP der Serie C und höher.* Erweiterter Status-Fehlercode vom Erweiterungs-E/A-Kommunikationsmodul. | | | |
| 24 | | *Nur für MicroLogix 1500 1764-LRP der Serie C und höher.* Zusätzliche Datenadresse auf Abfragepfad: Bits 7 bis 0: Ausgangselement, Bits 15 bis 8: Filenummer | | | |

(1) Der Anwenderzugriff bezieht sich auf den Anwenderprogrammzugriff (MSG-Filewort oder -Bit, das als Operand für einen Befehl in einem Kontaktplanlogik-Programm verwendet wird) oder auf den Zugriff über die Kommunikationsfunktion, wenn sich das System nicht im Herunterlade-Modus befindet (über die Programmiersoftware oder das Speichermodul).

Die Zielfileinformationen in den Unterelementen 12 bis 15 des MSG-Fileelements hängen vom Nachrichtentyp ab (siehe die nachfolgenden Tabellen).

| Informationen zur Zielposition des Nachrichtenfiles<br>Zielgerät = 485 CIF | | | | | |
|---|---|---|---|---|---|
| **Unter-element** | **Name** | **Beschreibung** | **Parameter** | **Größe** | **Anwender-programm-zugriff** |
| 12 | | Reserviert | J | Wort | Nur Lesen |
| 13 | MG11:0.TFN | Zielfilenummer | J | Wort | Lesen/Schreiben |
| 14 | MG11:0.ELE | Offset in Elementen im CIF | J | Wort | Lesen/Schreiben |
| 15 | | Reserviert | J | Wort | Nur Lesen |

| Informationen zur Zielposition des Nachrichtenfiles<br>Zielgerät = 500CPU oder PLC 5 | | | | | |
|---|---|---|---|---|---|
| **Unter-element** | **Adresse** | **Beschreibung** | **Parameter** | **Größe** | **Anwender-programm-zugriff** |
| 12 | | Zielfiletyp | J | Wort | Nur Lesen |
| 13 | MG11:0.TFN | Zielfilenummer[1] | J | Wort | Lesen/Schreiben |
| 14 | MG11:0.ELE | Zielfile-Elementnummer für B-, S-, N-, F-[2], T-, C-, R-, L-, ST- und RTC-[3]Files; oder Zielfile-Steckplatznummer für O- und I-Files. | J | Wort | Lesen/Schreiben |
| 15 | | Zielfile-Elementnummer für O- und I-Files. Für andere Files (außer O- oder I-Files) auf Null setzen. | J | Wort | Nur Lesen |

(1) Die Filenummer für RTC-Funktionsfiles wird durch die Programmiersoftware auf 0 gesetzt.

(2) Der F-File ist nur im MSG-Befehl für MicroLogix 1200- und 1500-Steuerungen der Serie C und höher zulässig.

(3) RTC und ST sind im MSG-Befehl nur für die MicroLogix 1200- und 1500-Steuerungen der Serie B und höher zulässig.

| **Informationen zur Zielposition des Nachrichtenfiles**<br>**Zielgerät = CIP-generisch**<br>***Nur für MicroLogix 1500 1764-LRP-Prozessoren der Serie C und höher.*** | | | | | |
|---|---|---|---|---|---|
| **Unter-element** | **Name** | **Beschreibung** | **Parameter** | **Größe** | **Anwender-programm-zugriff** |
| 12 | | Zielklasse | J | Wort | Nur Lesen |
| 13 | MG11:0.TFN | Zielinstanz | J | Wort | Lesen/ Schreiben |
| 14 | MG11:0.ELE | CIP-Datenanzahl senden | J | Wort | Lesen/ Schreiben |
| 15 | | Reserviert | J | Wort | Nur Lesen |

Die Steuerungs-Bits, Unterelement 16, des MSG-Fileelements werden nachfolgend beschrieben:

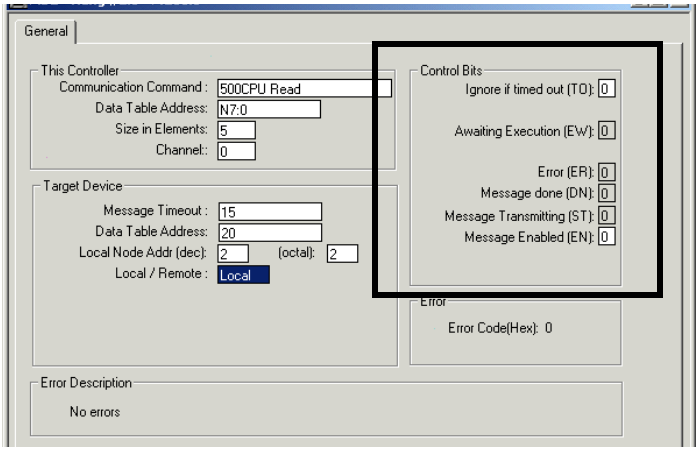
| **Nachrichtenfile-Unterelement 16 - Steuer-Bits** | | | | | |
|---|---|---|---|---|---|
| **Bit** | **Adresse** | **Beschreibung** | **Parameter** | **Größe** | **Anwender-programm-zugriff** |
| 15 | MG11:0.0/EN | Aktivieren<br>1=MSG aktiviert<br>0=MSG nicht aktiviert | N | Bit | Lesen/ Schreiben |
| 9 bis 14 | | Reserviert | N | Bit | Lesen/ Schreiben |
| 8 | MG11:0.0/TO | Zeitablauf<br>1=MSG-Zeitablauf durch Anwender<br>0=Nicht anwenderbedingter MSG-Zeitablauf | N | Bit | Lesen/ Schreiben |
| 0 bis 7 | | Reserviert | N | Bit | Lesen/ Schreiben |

Die Status-Bits, Unterelement 17, des MSG-Fileelements werden nachfolgend beschrieben.

**Nachrichtenfile-Subelement 17 – Status-Bits**

| Bit | Adresse | Beschreibung | Parameter | Größe | Anwender-programm-zugriff |
|---|---|---|---|---|---|
| 15 | | Reserviert | N | Bit | Nur Lesen |
| 14 | MG11:0.0/ST | Start:<br>1 = MSG übertragen und durch Zielgerät bestätigt<br>0 = MSG wurde von Ziel nicht empfangen | N | Bit | Nur Lesen |
| 13 | MG11:0.0/DN | Fertig<br>1 = MSG erfolgreich abgeschlossen<br>0 = MSG nicht abgeschlossen | N | Bit | Nur Lesen |
| 12 | MG11:0.0/ER | Fehler<br>1 = Fehler erkannt<br>0 = Kein Fehler erkannt | N | Bit | Nur Lesen |
| 11 | | Reserviert | N | Bit | Nur Lesen |
| 10 | MG11:0.0/EW | Aktiviert und wartend<br>1=MSG aktiviert und wartend<br>0=MSG nicht aktiviert und wartend | N | Bit | Nur Lesen |
| 1 bis 9 | | Reserviert | N | Bit | Nur Lesen |
| 0 | MG11:0.0/R | Für PCCC-Nachrichten:<br>Bereich (1 = zentral, 0 = dezentral)<br>Für CIP-Nachrichten: Ziel (1 = Kommunikationsmodul, 0 = Netzwerkgerät) | J | Bit | Nur Lesen |

## „Control Bits"- Parameter (Steuer-Bits-Parameter)




*Bei Zeitablauf ignorieren (TO)*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| MG11:0/TO | Binärwert | Ein oder aus | Steuerung | Lesen/Schreiben |

Das TO-Bit (Zeitablauf) kann in einer Anwendung gesetzt werden, um einen aktiven Nachrichtenbefehl aus der Prozessorsteuerung zu ent- fernen. Durch Überwachung des EW- und des ST-Bits zum Starten eines Zeitwerks kann eine Zeitablauf-Routine erstellt werden. Bei Zeitablauf des Zeitwerks kann das TO-Bit gesetzt werden, wodurch die Nachricht aus dem System entfernt wird. Die Steuerung setzt das TO-Bit zurück, sobald der zugehörige MSG-Strompfad das nächste Mal eine Statuswechsel von unwahr nach wahr verzeichnet.

Eine einfachere Methode ist die Verwendung der Nachrichten-Zeitablauf-Variablen (Beschreibung siehe Seite 21-23), da dies das Anwenderprogramm vereinfacht. Diese integrierte Steuerung des Nachrichten-Zeitablaufs ist aktiv, sobald der Nachrichten-Zeitablauf ungleich null ist. Der Standardwert beträgt 5 Sekunden, d. h. die interne Zeitablauf-Steuerung wird automatisch aktiviert, wenn der Standardwert nicht geändert wird.

Bei Verwendung des internen Zeitablaufs und Unterbrechung der Kommunikation wird nach Ablauf der festgelegten Zeitspanne der MSG-Befehl unterbrochen und ein Fehler generiert. Somit kann ein Wiederholungsversuch durchgeführt oder gegebenenfalls eine andere Maßnahme ergriffen werden.

Setzen Sie die Variable für den MSG-Nachrichtenzeitablauf auf null, um die interne Zeitablauf-Steuerung zu deaktivieren. In diesem Fall wartet der Prozessor bei einer Unterbrechung der Kommunikation für eine unbestimmte Zeit auf eine Antwort. Wenn durch Setzen des ST-Bits der Empfang einer Bestätigungsmeldung (ACK) angezeigt wird, während die Antwort selbst nicht empfangen wurde, scheint der MSG-Befehl blockiert zu sein; tatsächlich jedoch wartet der Befehl nach wie vor auf eine Antwort von dem Zielgerät.

*Aktivieren (EN)*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| MG11:0/EN | Binärwert | Ein oder aus | Steuerung | Lesen/Schreiben |

Das Freigabe-Bit (EN) wird gesetzt, wenn der Strompfad wahr und damit der MSG-Befehl aktiviert wird. Der MSG-Befehl wird aktiviert, wenn das Befehlspaket erstellt und in einen der MSG-Puffer gestellt oder die Nachrichtenaufforderung in die MSG-Warteschlange einge- reiht wurde. Dieses Bit bleibt gesetzt, bis die Nachrichtenübertragung abgeschlossen ist und der Strompfad unwahr wird. Dieses Bit kann gelöscht werden, wenn das ER- oder das DN-Bit gesetzt ist, um bei der nächsten Abfrage bei einem wahren Strompfad erneut einen MSG-Befehl auszulösen.

**WICHTIG** Dieses Bit darf nicht über das Steuerprogramm gesetzt werden.

*Aktiviert und warten (EW)*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| MG11:0/EW | Binärwert | Ein oder aus | Status | Nur Lesen |

Das EW-Bit (Enabled und Waiting) wird gesetzt, nachdem das Aktivierungs-Bit gesetzt wurde und sich die Nachricht im Puffer (nicht in der Warteschlange) befindet und zum Senden bereit ist. Das EW-Bit wird gelöscht, wenn die Nachricht gesendet wurde und der Prozessor eine Bestätigung (ACK) vom Zielgerät empfängt. Dies geschieht, bevor das Zielgerät die Nachricht verarbeitet hat und eine Antwort senden konnte.

*Fehler (ER)*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| MG11:0/ER | Binärwert | Ein oder aus | Status | Nur Lesen |

Das ER-Bit (Fehler) wird gesetzt, wenn die Nachrichtenübertragung nicht erfolgreich durchgeführt wurde. Ein Fehlercode wird in dem MSG-File abgelegt. Das ER-Bit und der Fehlercode werden gelöscht, wenn der zugehörige Strompfad von unwahr nach wahr wechselt.

*Fertig (DN)*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| MG11:0/DN | Binärwert | Ein oder aus | Status | Nur Lesen |

Das DN-Bit (Fertig) wird nach erfolgreicher Übertragung der Nach- richt gesetzt. Das DN-Bit wird gelöscht, wenn der dazugehörige Strompfad das nächste Mal von unwahr nach wahr wechselt.

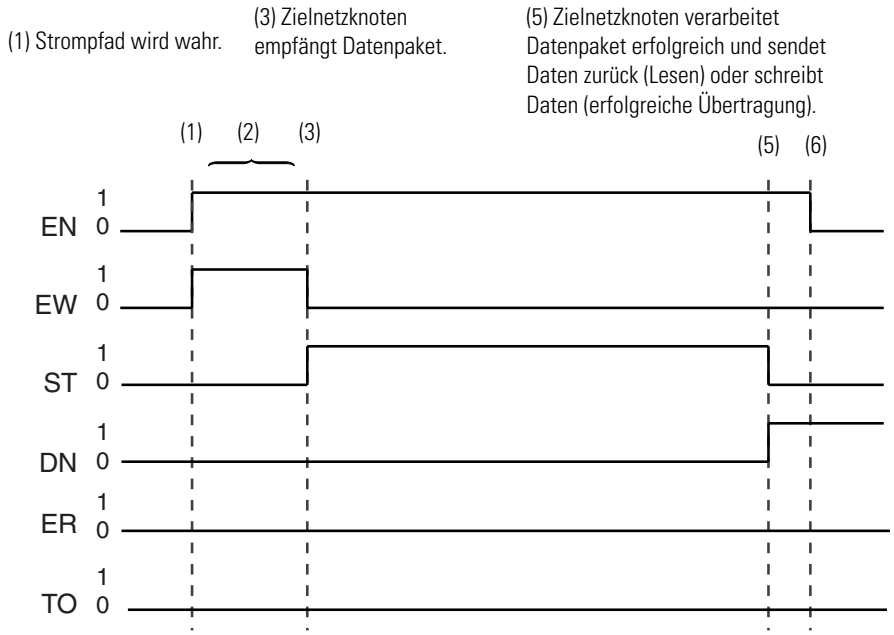
*Start (ST)*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| MG11:0/ST | Binärwert | Ein oder aus | Status | Nur Lesen |

Das Start-Bit (ST) wird gesetzt, wenn der Prozessor eine Bestätigung (ACK) vom Zielgerät empfängt. Das ST-Bit wird gelöscht, wenn das DN-, ER- oder TO-Bit gesetzt wird.

## Zeitdiagramm für MSG-Befehl

In den folgenden Abschnitten wird das Zeitdiagramm eines Nachrichtenbefehls beschrieben.




**1.** Falls in einem der vier aktiven Nachrichtenpuffer Platz ist, wenn der MSG-Strompfad wahr wird und der MSG-Befehl abgefragt wird, werden die EN- und EW-Bits für diese Nachricht gesetzt. Bei einem MSG-Schreibbefehl werden die Quellendaten zu diesem Zeitpunkt in den Nachrichtenpuffer gestellt.

(Nicht im Diagramm dargestellt.) Wenn in keinem der vier Nachrichtenpuffer Speicherplatz verfügbar ist, wird die Nachrichtenaufforderung in die MSG-Warteschlange gestellt und nur das EN-Bit gesetzt. Die MSG-Warteschlange basiert auf dem FIFO-Prinzip, damit die Reihenfolge, in der die MSG-Befehle aktiviert wurden, in der Steuerung gespeichert werden kann. Die erste Nachricht in der Warteschlange wird in den nächsten frei werdenden Puffer gestellt, und das EW-Bit wird gesetzt (1).

**HINWEIS** Das Steuerprogramm hat keinen Zugriff auf die Nachrichtenpuffer oder die Kommunikationswarteschlange.

Das EN-Bit bleibt gesetzt (1), bis der gesamte Nachrichtenübertragungsvorgang abgeschlossen ist und das DN-, das ER- oder das TO-Bit gesetzt wird (1). Die Zeitmessung des MSG-Nachrichten- ablaufs beginnt, sobald das EN-Bit gesetzt wird (1). Wenn der Zeitablauf vor Abschluss des MSG-Befehls eintritt, wird das ER-Bit gesetzt (1), und ein Fehlercode (37H) wird als Hinweis auf den Zeitablauffehler in den MG-File gestellt.

**2.** Am Ende der nächsten Abfrage oder des nächsten REF- oder SVC-Befehls entscheidet die Steuerung, ob die Kommunikations-warteschlange auf weitere Befehle überprüft werden soll. Die Steuerung macht ihre Entscheidung vom Zustand der CSS- (Communication Servicing Selection) und MSS-Befehle (Message Servicing Selection) des Kanals, den Netzwerk-Kommunikations-, anforderungen anderer Netzknoten und davon abhängig, ob vorherige Nachrichtenbefehle bereits verarbeitet werden. Wenn die Warteschlange nicht erneut überprüft wird, behält der MSG-
Befehl seinen Status bei. Das EN-Bit und das EW-Bit bleiben gesetzt (1) oder nur das EN-Bit bleibt bis zum Ende der nächsten Abfrage bzw. des nächsten REF- oder SVC-Befehls gesetzt.

Wenn die Steuerung in der Warteschlange einen Befehl erkennt, werden die Einträge in der Kommunikationswarteschlange in die Nachrichtenpuffer gestellt, bis alle vier Nachrichtenpuffer belegt sind. Beim Entfernen einer ungültigen Nachricht aus der Kommu-nikationswarteschlange wird das ER-Bit im MG-File gesetzt (1). Außerdem wird ein Code in den MG-File geschrieben, um den Anwender auf den Fehler hinzuweisen. Wenn ein gültiger Nach- richtenbefehl in einen Nachrichtenpuffer geladen wird, werden die EN- und EW-Bits für diese Nachricht gesetzt (1).

Dann verlässt die Steuerung das Abfrageende oder den REF- oder SVC-Befehl der Abfrage. Die HintergrundKommunikationsfunktion der Steuerung sendet die zu übertragenden Nachrichten an die Zielnetzknoten, die im Nachrichtenbefehl angegeben wurden. Abhängig vom Zustand der CSS- und MSS-Bits können Sie jeder- zeit bis zu vier aktive Nachrichtenbefehle pro Kanal bedienen.

**3.** Wenn der Zielnetzknoten die Nachricht erfolgreich empfängt, wird eine Bestätigung (ACK) zurückgesendet. Aufgrund der Bestätigung setzt der Prozessor das EW-Bit zurück (0) und setzt (1) das ST-Bit. Beachten Sie bitte, dass der Zielnetzknoten zu diesem Zeitpunkt das Datenpaket noch nicht überprüft hat, um festzustellen, ob er die Anforderung bearbeiten kann.

Nachdem das ST-Bit gesetzt wurde (1), wartet die Steuerung auf eine Antwort von dem Zielnetzknoten. Der Zielnetzknoten muss nicht innerhalb eines bestimmten Zeitrahmens antworten.

**HINWEIS** Wenn an dem Zielnetzknoten während der Nach-richtenübertragung ein Fehler auftritt oder die Spannungsversorgung aus- und wieder eingeschal- tet wird, sendet der Zielnetzknoten keine Antwort. Aus diesem Grund sollte in dem MSG-Befehl ein Wert für den Nachrichten-Zeitablauf festgelegt werden.

4. Schritt 4 wird im Zeitdiagramm nicht dargestellt. Wenn keine Bestätigungsnachricht (ACK) empfangen wird, wird Schritt 3 nicht ausgeführt. Stattdessen wird entweder keine Antwort oder eine negative Bestätigung (NAK) empfangen. In diesem Fall bleibt das STBit zurückgesetzt (0).

   Mögliche Ursachen für das Ausbleiben einer Antwort:
   - der Zielnetzknoten ist nicht vorhanden
   - die Nachricht wurde bei der Übertragung beschädigt
   - die Antwort wurde bei der Übertragung beschädigt

   Mögliche Ursachen für NAK (keine Bestätigung):
   - der Zielnetzknoten ist besetzt
   - der Zielnetzknoten hat eine beschädigte Nachricht erhalten
   - die Nachricht ist zu lang.

   Bei einer NAK bleiben das EW-Bit zurückgesetzt (0) und das ER-Bit gesetzt (1) als Hinweis darauf, dass der Nachrichtenbefehl nicht erfolgreich durchgeführt wurde.

5. Nach erfolgreichem Empfang des Pakets sendet der Zielnetzknoten ein Antwortpaket. Das Antwortpaket enthält eine der folgenden Antworten:
   - erfolgreiche Schreibaufforderung
   - erfolgreiche Leseaufforderung mit Daten
   - Fehler mit Fehlercode

   Am Ende der nächsten Abfrage bzw. des nächsten REF- oder SVC-Befehls und nach Empfang der Antwort des Zielnetzknotens prüft die Steuerung die von dem Zielgerät erhaltene Nachricht. Wenn die Antwort erfolgreich ist, werden das DN-Bit gesetzt (1) und das ST-Bit zurückgesetzt (0). Bei einer erfolgreichen Leseaufforderung werden die Daten in die Datentafel geschrieben. Die Nachrichten- befehlsfunktion ist somit abgeschlossen.

   Wenn die Antwort einen Fehler mit einem Fehlercode enthält, werden das ER-Bit gesetzt (1) und das ST-Bit zurückgesetzt (0). Die Nachrichtenbefehlsfunktion ist somit abgeschlossen.

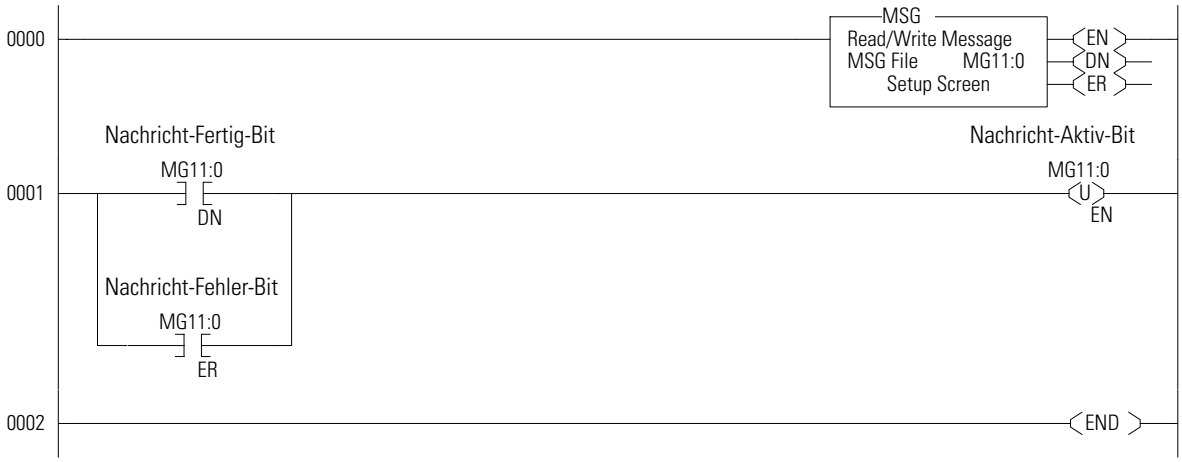
6. Wenn das DN-Bit oder das ER-Bit gesetzt (1) und der MSG-Strompfad unwahr ist, wird das EN-Bit bei der nächsten Abfrage des Nachrichtenbefehls zurückgesetzt (0).

Beispiele zur Verwendung des Nachrichtenbefehls finden Sie im Abschnitt „Kontaktplanlogik für MSG-Befehl“ auf Seite 21-15.

# Kontaktplanlogik für MSG-Befehl

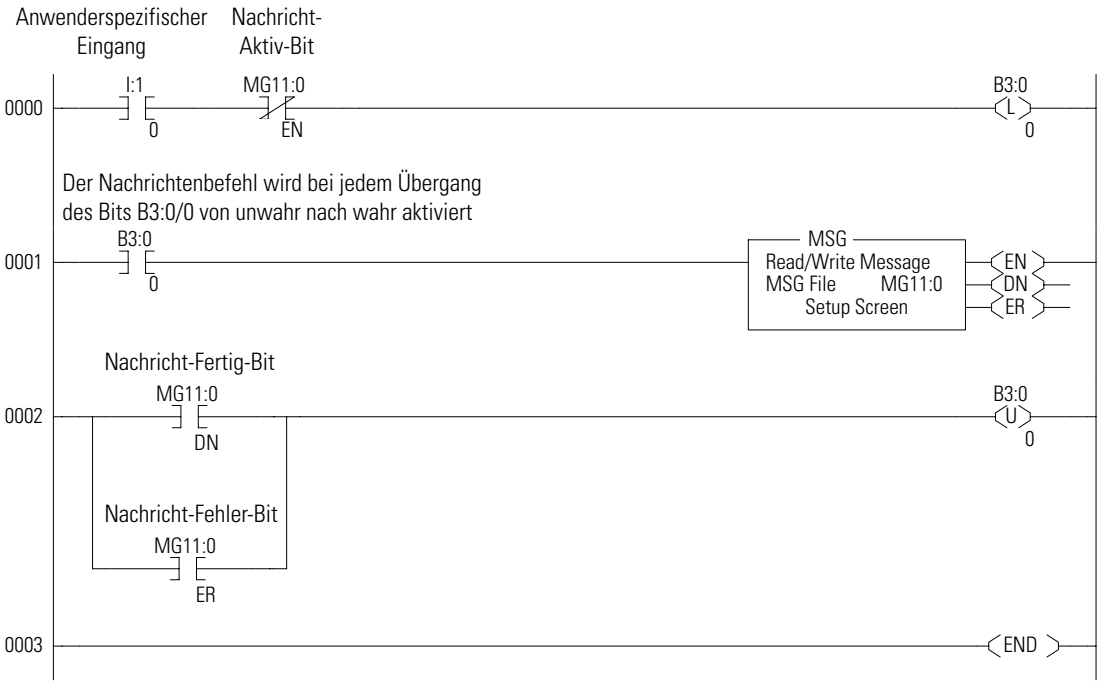
## MSG-Befehl für kontinuierlichen Betrieb aktivieren

Der Nachrichtenbefehl wird bei der erstmaligen Programmabfrage durch den Prozessor sowie nach jeder vollständigen Durchführung der Nachricht aktiviert, beispielsweise nach dem Setzen des DN- oder des ER-Bits.




## MSG-Befehl über Anwendereingang aktivieren

Dieses Beispiel zeigt das Steuerungsprinzip während der Ausführung eines Nachrichtenbefehls. Der Eingang I:1/0 könnte ein beliebiges, vom Anwender eingegebenes Bit sein, das den Zeitpunkt steuert, an dem die Nachrichten gesendet werden. Sobald I:1/0 gesetzt und die Nachricht MG11:0 *nicht aktiviert* ist, wird der Nachrichtenbefehl auf Strompfad 0001 aktiviert.



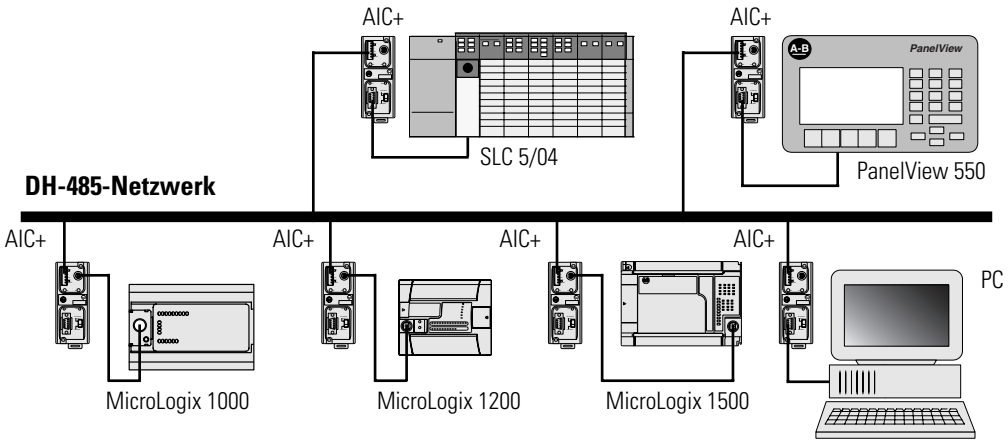
# Zentrale Nachrichten

Über die Steuerung können sowohl zentrale als auch dezentrale Nachrichten übertragen werden. Bei einer zentralen Nachricht sind alle Geräte ohne ein zusätzliches, als Brücke verwendetes Gerät zugänglich. Für den Anschluss an das Netzwerk sind unter Umständen verschiedene elektrische Schnittstellen erforderlich, doch das Netz- werk wird dennoch als zentrales Netzwerk betrachtet. Dezentrale Nachrichten verwenden ein dezentrales Netzwerk, wobei auf die Geräte nur über ein zusätzliches Gerät (Durchlaufen oder Routing) zugegriffen werden kann. Eine ausführliche Beschreibung zu dezentralen Netzwerken finden Sie auf Seite 21-37.

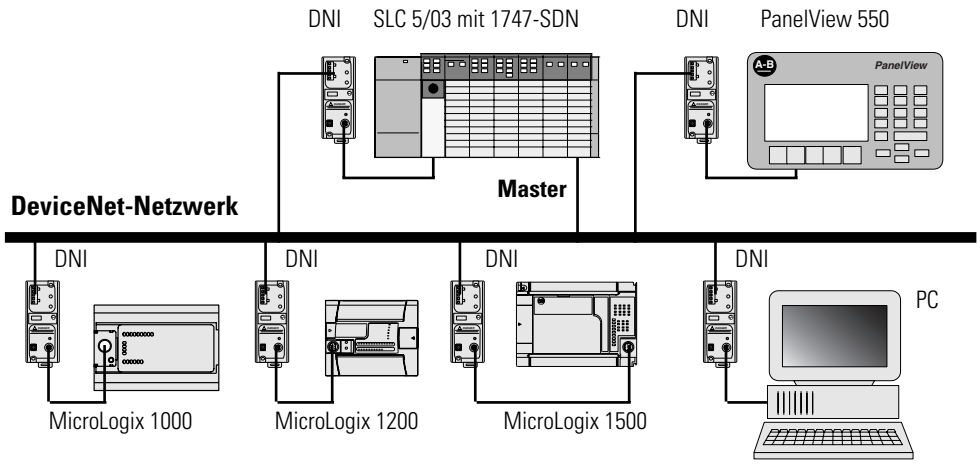
## Zentrale Netzwerke

Die folgenden drei Beispiele stellen unterschiedliche Typen lokaler und dezentraler Netzwerke dar.

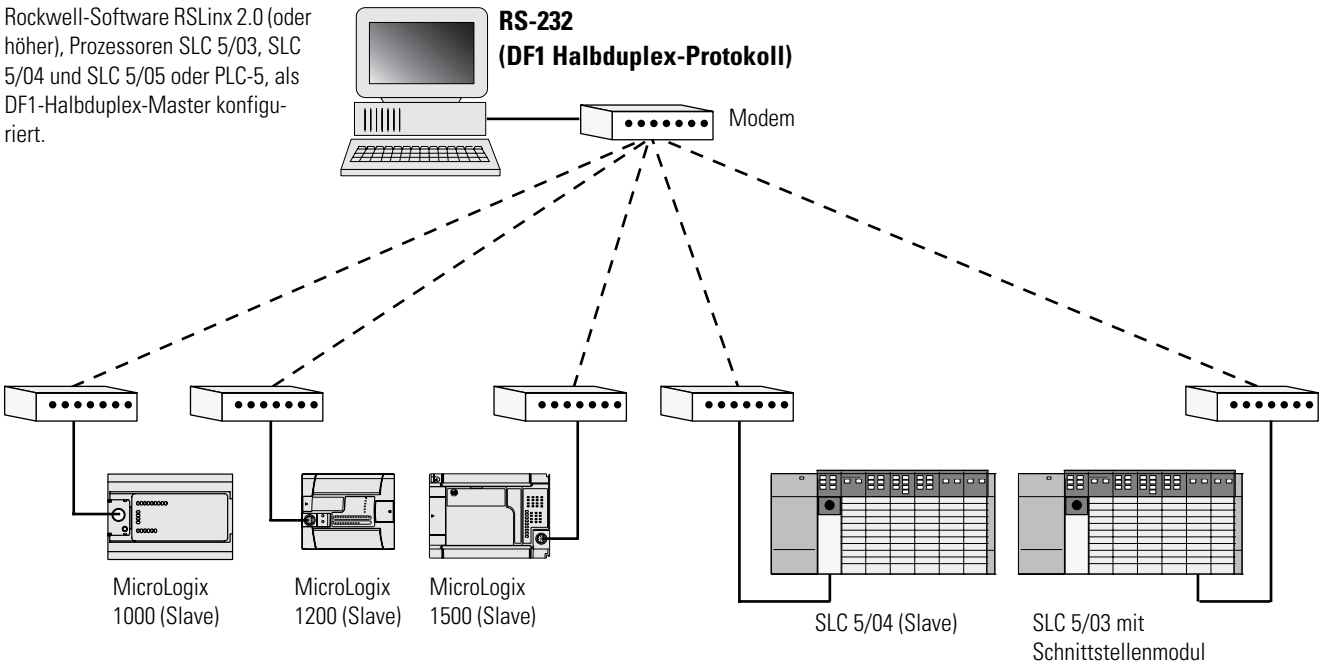
*Beispiel 1 - Lokales DH-485-Netzwerk mit AIC+ (1761-NET-AIC)-Schnittstelle*




*Beispiel 2 - zentrales DeviceNet-Netzwerk mit DeviceNet-Schnittstelle (1761-NET-DNI)*



*Beispiel 3 - zentrales DF1-Halbduplex-Netzwerk*

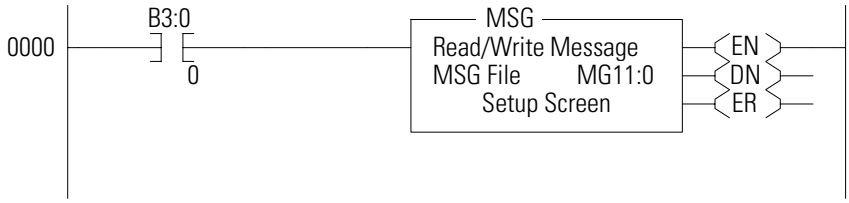

Rockwell-Software RSLinx 2.0 (oder höher), Prozessoren SLC 5/03, SLC 5/04 und SLC 5/05 oder PLC-5, als DF1-Halbduplex-Master konfiguriert.
RS-232
(DF1 Halbduplex-Protokoll)
Modem
MicroLogix 1000 (Slave)
MicroLogix 1200 (Slave)
MicroLogix 1500 (Slave)
SLC 5/04 (Slave)
SLC 5/03 mit Schnittstellenmodul

**HINWEIS** Zwischen der Steuerung und dem Modem wird eine Isolierung (1761-NET-AIC) empfohlen.

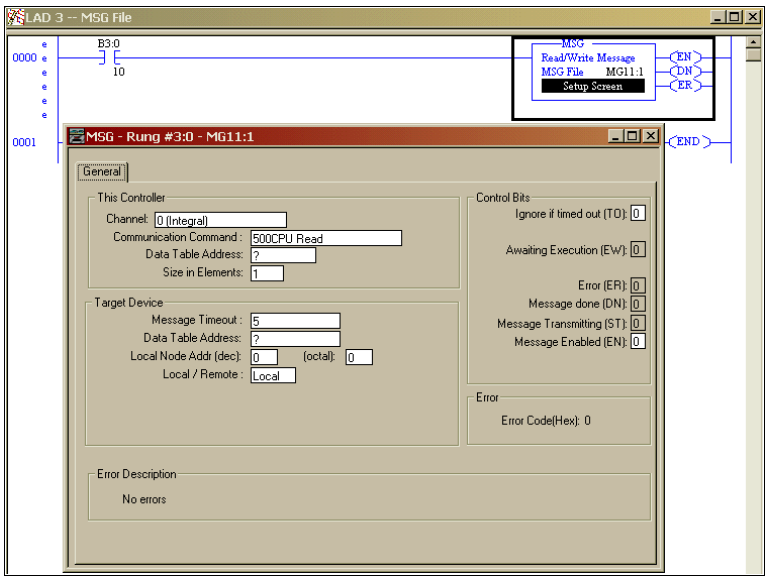
# Zentrale Nachrichtenübertragung konfigurieren

## Bildschirm Nachrichtenkonfiguration

Der nachfolgende Strompfad zeigt einen MSG-Befehl mit vorangehen- der bedingter Logik. Doppelklicken Sie auf „*Setup Screen*“ (Konfigurationsbildschirm), um den Bildschirm „Message Setup“ (Nachrichtenkonfiguration) zu öffnen.



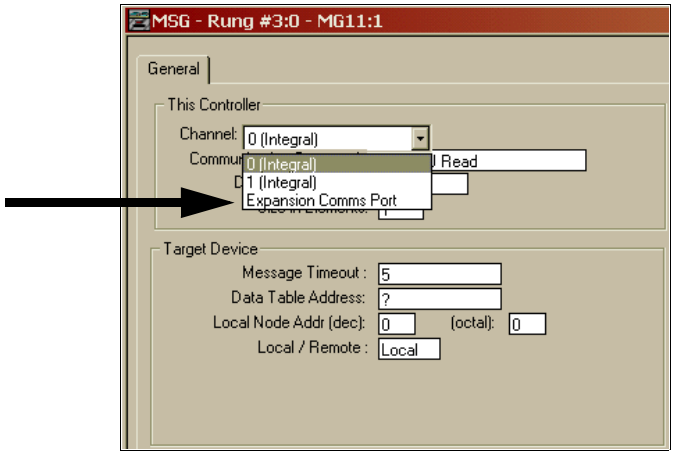

Diese Abbildung zeigt den RSLogix-Einrichtungsbildschirm für die Nachrichtenübertragung. Über diesen Bildschirm können Sie die Elemente „This Controller“ (Diese Steuerung), „Target Device“ (Zielgerät) und „Control Bits“ (Steuerbits) konfigurieren. Im Nachfogenden werden diese Elemente näher beschrieben.



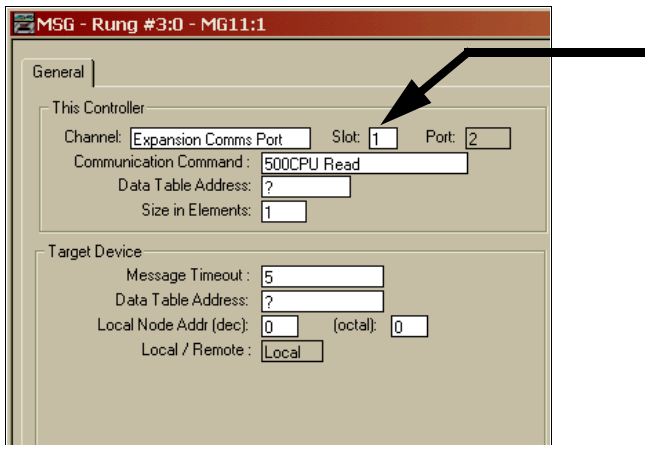
## Paramater für „This Controller“ (Diese Steuerung)

*Kanal*

MicroLogix 1200 und MicroLogix 1500 1764-LSP unterstützen nur Nachrichtenübermittlung über Kanal 0. MicroLogix 1500 1764-LRP unterstützt drei verschiedene Arten der Nachrichtenübertragung. Kanäle 0 und 1 sind RS-232-Anschlüsse und entsprechen ihrer Funktion nach Kanal 0 der MicroLogix 1200- und MicroLogix 1500-Steuerungen 1764-LSP. Der Prozessor 1764-LRP unterstützt zudem Backplane-Kommunikation durch den Erweiterungs-Kommunikationsanschluss (ECP - Expansion Communication Port), wie im Folgenden dargestellt wird.

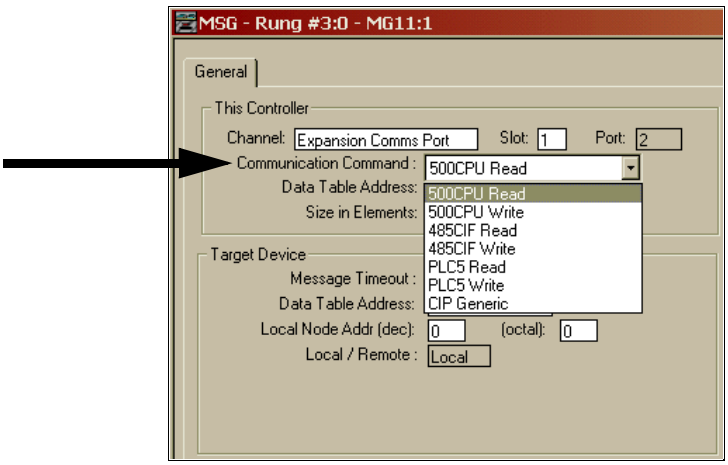



Wenn der ECP (Expansion Communication Port) ausgewählt wird, können Sie wählen, an welcher Steckplatzposition (1 bis 16) sich der Scanner befinden soll. Der 1764-LRP-Prozessor kann bis zu zwei 1769-SDN-Scannermodule mit voller Nachrichtenfunktion unterstützen.




**HINWEIS** Sie können mehrere 1769-SDN-Scannermodule in einem 1764-LRP MicroLogix 1500-System verwenden, jedoch nur über die ersten beiden Module Nachrichten versenden. Ein Scanner, der sich physikalisch hinter den ersten beiden befindet, kann nur zum E/A-Scannen verwendet werden.

*Kommunikationsbefehl*




Die Steuerung unterstützt sechs verschiedene Typen an Kommunikationsbefehlen (für MicroLogix 1500 1764-LRP Serie C und höher sind es sieben Typen).Wenn das Zielgerät einen dieser Befehlstypen unterstützt, sollte die Steuerung in der Lage sein, Daten mit diesem Gerät auszutauschen. Unterstützte Befehle:

**Tabelle 21.4 Kommunikationsbefehltypen**

| Kommunikationsbefehl | Beschreibung | Zweck |
|---|---|---|
| 500CPU-Lesebefehl | Das Zielgerät ist kompatibel mit dem SLC 500-Befehlssatz (alle MicroLogix-Steuerungen) und unterstützt diesen Befehlssatz. | Daten lesen |
| 500CPU-Schreibbefehl | Das Zielgerät ist kompatibel mit dem SLC 500-Befehlssatz (alle MicroLogix-Steuerungen) und unterstützt diesen Befehlssatz. | Daten senden |
| 485CIF-Lesebefehl(1) | Das Zielgerät ist kompatibel mit dem 485CIF-Befehlssatz (PLC2) und unterstützt diesen Befehlssatz. | Daten lesen |
| 485CIF-Schreibbefehl(1) | Das Zielgerät ist kompatibel mit dem 485CIF-Befehlssatz (PLC2) und unterstützt diesen Befehlssatz. | Daten senden |
| PLC5-Lesebefehl | Das Zielgerät ist kompatibel mit dem PLC5-Befehlssatz und unterstützt diesen Befehlssatz. | Daten lesen |
| PLC5-Schreibbefehl | Das Zielgerät ist kompatibel mit dem PLC5-Befehlssatz und unterstützt diesen Befehlssatz. | Daten senden |
| CIP-generisch(2) | Das Zielgerät ist kompatibel mit dem CIP-Befehlssatz für DeviceNet und unterstützt diesen Befehlssatz. | Daten senden und empfangen |

(1) Siehe den Hinweis „Wichtig“ unten.

(2) MicroLogix 1500 1764-LRP der Serie C und höher nur für DeviceNet-Nachrichtenübermittlung.

**WICHTIG** Der CIF-File (Common Interface File) in den MicroLogix 1200-, 1500- und SLC 500-Prozessoren ist File 9. Der CIF-File in der MicroLogix 1000-Steuerung ist Ganzzahl-File 7.

### *Datentafeladresse*

Diese Variable bezeichnet die Startadresse in der zentralen Steuerung. Die gültigen Filetypen für die Datentafeladresse werden nachfolgend beschrieben.

| **Nachricht lesen** | **Nachricht schreiben** |
|---|---|
| Bit (B) | Ausgang (O) |
| Zeitwerk (T) | Eingang (I) |
| Zähler (C) | Bit (B) |
| Steuerung (R) | Zeitwerk (T) |
| Ganzzahl (N) | Zähler (C) |
| Fließkomma (F)(1) | Steuerung (R) |
| Doppelwort (L) | Ganzzahl (N) |
| | Fließkomma (F)(1) |
| | Doppelwort (L) |
| | Zeichenkette (ST)(2)(3) |
| | Echtzeituhr (RTC)(2)(4) |

(1) Gilt nur für MicroLogix 1200 Serie C and höher sowie MicroLogix 1500 Serie C und höher. Der Nachrichtentyp muss 500CPU oder PLC5 sein. Der zentrale Filetyp und der Zielfiletyp müssen beide Fließkommafiles sein.

(2) Gilt nur für MicroLogix 1200 Serie B und höher sowie 1500 Serie B und höher.

(3) 485CIF-Schreibbefehl (nur ST-zu-485CIF).

(4) 500CPU-Schreibbefehl (nur RTC zu Ganzzahl oder RTC zu RTC).

### *Größe in Elementen*

Diese Variable gibt das Datenvolumen (Anzahl der Elemente) an, die mit dem Zielgerät ausgetauscht werden sollen.

Mit einem MSG-Befehl können maximal 103 Worte (206 Bytes) übertragen werden; dieses maximale Volumen wird durch den Zieldatentyp bestimmt. Der Zieldatentyp wird durch den Nachrich- tentyp festgelegt: Lese- oder Schreibnachricht

- Lesenachrichten: Bei Lesenachrichten ist der Datenfile in dem zentralen oder initiierenden Prozessor der Zielfile.

  **HINWEIS** Die Filetypen für Eingang, Ausgang, Zeichenkette und RTC gelten nicht für Lesenachrichten.

- Schreibnachrichten: Bei Schreibnachrichten ist der Datenfile in dem Zielprozessor der Zielfile.

Die maximale Anzahl von Elementen, die übertragen oder empfangen werden können, entnehmen Sie bitte der nachfolgenden Tabelle. Dabei können die verschiedenen Filetypen nicht beliebig verwendet werden. Beispielsweise ist es nicht möglich, ein Zeitwerk in einen Ganzzahl-File einzulesen oder Zähler in einen Zeitwerkfile zu schreiben. Ausnahmen dieser Regel:

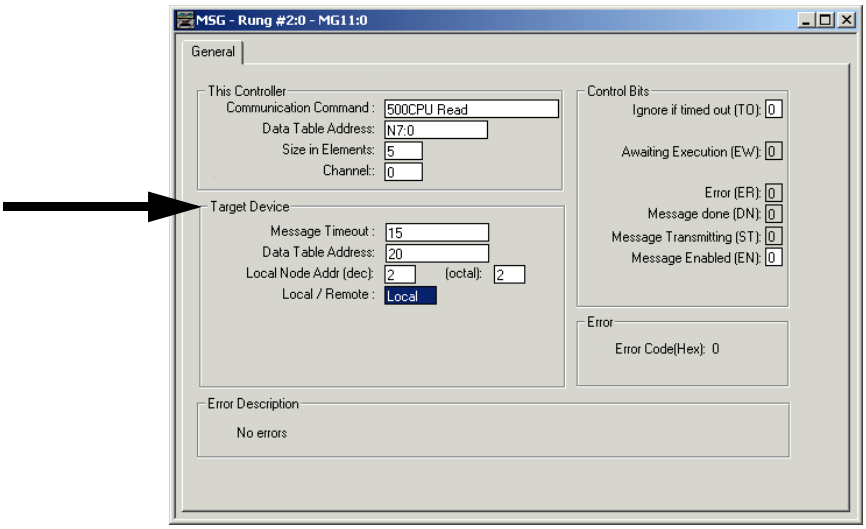
- Lange ganzzahlige Daten können aus Bit- oder Ganzzahl-Files gelesen oder in diese geschrieben werden. Und
- RTC-Files können in Ganzzahl-Files geschrieben werden (*nur MicroLogix 1200 Serie B und höher sowie MicroLogix 1500 Serie B und höher*).

**HINWEIS** Die nachfolgende Tabelle soll nicht die Kompatibilität der verschiedenen Filetypen darstellen, sondern nur die maximale Anzahl der Elemente, die in den jewei- ligen Fällen ausgetauscht werden können.

| Nachrichtentyp | Filetyp | Elementgröße | Maximale Anzahl Elemente pro Nachricht |
|---|---|---|---|
| 485CIF | O, I, B, N | 1-Wort | 103 |
| | L | 2-Wort | 51 |
| | T, C, R | 3-Wort | 34 |
| | ST | 42-Wort | 2 (nur Schreiben) |
| 500CPU | O, I, B, N | 1-Wort | 103 |
| | F(1), L | 2-Wort | 51 |
| | T, C, R | 3-Wort | 34 |
| | RTC | 8-Wort | 1 (nur Schreiben) |
| PLC5 | O, I, B, N | 1-Wort | 103 |
| | F(1), L | 2-Wort | 51 |
| | T | 5-Wort | 20 |

(1) Gilt nur für MicroLogix 1200 Serie C and höher sowie MicroLogix 1500 Serie C und höher. Der Nachrichtentyp muss 500CPU oder PLC5 sein. Der zentrale Filetyp und der Zielfiletyp müssen beide Fließkommafiles sein.

## Parameter für „Target Device“ (Zielgerät)




### *Nachrichten-Zeitablauf*

Dieser Wert legt die Zeitspanne (in Sekunden) fest, die vom Start bis zum Abschluss eines Nachrichtenbefehls zur Verfügung stehen. Die Zeitmessung beginnt, wenn der Strompfadstatus von unwahr nach wahr wechselt und damit die Nachrichtenübertragung möglich wird. Nach Ablauf dieser Zeitspanne tritt ein Nachrichtenfehler auf. Der Standardwert ist 5 Sekunden. Der maximale Wert für den Zeitablauf ist 255 Sekunden.

Wird der Nachrichten-Zeitüberlauf auf null gesetzt, kommt es in kei- nem Fall zu einem Nachrichtenfehler wegen Zeitablauf. Setzen Sie das Zeitablauf-Bit (TO = 1) so, dass ein Nachrichtenbefehl aus dem Speicher gelöscht wird, wenn das Zielgerät nicht auf die Kommuni- kationsaufforderung reagiert.

### *Datentafeladresse/-Offset*

Diese Variable bezeichnet die Startadresse in der Zielsteuerung. Die Datentafeladresse wird bei 500CPU- und PLC5-Nachrichten verwen- det. Jeglicher konfigurierter Datenfile innerhalb des Zielgeräts, dessen Filetyp von der Steuerung erkannt wird, kann als gültige Datenadresse verwendet werden. Gültige Kombinationen sind nachfolgend aufgeführt:

| Nachrichtentyp | Zentraler Filetyp | Zielfiletyp |
|---|---|---|
| 500CPU und PLC5 | O, I, B, N, F(1), L | O, I, S, B, N, F(1), L |
| | T | T |
| | C | C |
| | r | r |
| | RTC(2) | N, RTC |

(1) Gilt nur für MicroLogix 1200 Serie C and höher sowie MicroLogix 1500 Serie C und höher. Der Nachrichtentyp muss 500CPU oder PLC5 sein. Der zentrale Filetyp und der Zielfiletyp müssen beide Fließkommafiles sein.

(2) 500CPU-Schreibbefehl (nur RTC zu Ganzzahl oder RTC zu RTC). Gilt nur für MicroLogix 1200 Serie B und höher sowie 1500 Serie B und höher

Der Datentafel-Offset wird bei 485CIF-Nachrichten verwendet. Als gültiger Offset kann jeder Wert in einem Bereich von 0 bis 255 ver- wendet werden; dieser Wert bezeichnet den Wort- oder Byte-Offset in dem CIF-File (Common Interface File) des Ziels. Ob es sich dabei um einen Wort- oder Byte-Offset handelt, hängt von dem jeweiligen Gerätetyp ab. MicroLogix-Steuerungen und SLC-Prozessoren verwen- den Wort-Offsets; PLC-5- und ControlLogix-Prozessoren verwenden Byte-Offset.

### *Zentrale Netzknotenadresse*

Dies ist die Netzknotennummer des Zielgeräts, wenn sich die Geräte in einem DH-485- (mit 1761-NET-AIC), DeviceNet- (mit 1761-NET-DNI) oder DF1-Halbduplex-Netzwerk befinden.

**HINWEIS** Zur Initiierung einer Rundsendenachricht auf einem DH-485-Netzwerk setzen Sie die lokale Netzknoten- adresse auf -1.

### *Zentral/dezentral*

Diese Variable bestimmt den zu verwendenden Kommunikationstyp. Verwenden Sie den lokalen Kommunikationstyp, wenn eine Punkt-zu-Punkt-Kommunikation über DF1-Vollduplex oder Netzwerkkommunikation wie DH-485 (mit 1761-NET-AIC), DeviceNet (mit 1761-NET-DNI) oder DF1-Halbduplex erwünscht ist.

## Beispiele für zentrale Nachrichten

Dieser Abschnitt enthält vier Beispiele zur zentralen Nachrichtenfunktion:

- Nachrichtentyp 500CPU
- Nachrichtentyp 485CIF
- Nachrichtentyp PLC5
- Nachrichtentyp CIP-generisch über DeviceNet

Eine Zusammenfassung der Konfigurationsparameter des Nachrichtenbefehls finden Sie nachfolgend.

| **Parameter** | | **Beschreibung** |
|---|---|---|
| This Controller (Diese Steuerung) | Kommunikationsbefehl | Gibt den Nachrichtentyp an. Gültige Nachrichtentypen:<br>• 500CPU-Lesebefehl<br>• 500CPU-Schreibbefehl<br>• 485CIF-Lesebefehl<br>• 485CIF-Schreibbefehl<br>• PLC5-Lesebefehl<br>• PLC5-Schreibbefehl |
| | Data Table Address (Datentafeladresse) | Bei einem Lesebefehl ist die die Startadresse für den Empfang von Daten.<br>Gültige Filetypen: B, T, C, R, N und L. |
| | | Bei einem Schreibbefehl ist dies die Startadresse, die an das Zielgerät gesendet wird.<br>Gültige Filetypen sind O, I, B, T, C, R, N, L, ST(1)(2) und RTC(2)(3). |
| | Size in Elements (Größe in Elementen) | Definiert die Länge der Nachricht in Elementen.<br>• 1-Wort Elemente; gültige Größe: 1 bis 103.<br>• 2-Wort Elemente; gültige Größe: 1 bis 51.<br>• 8-Wort Elemente; gültige Größe: 1<br>• 42-Wort Elemente; gültige Größe: 1 bis 2<br>• Zeitwerk- (500CPU und 485CIF), Zähler- und Steuerungselemente; gültige Größe: 1 bis 34.<br>• PLC-5-Zeitwerkelemente; gültige Größe: 1 bis 20 |
| | Channel (Kanal) | Bezeichnet den Kommunikationskanal. Immer Kanal 0 (*oder Kanal 1 – nur für MicroLogix 1500 1764-LRP-Prozessoren.*) |
| Target Device (Zielgerät) | Nachrichtenzeitablauf | Gibt an, wie lange die Steuerung auf die Antwort wartet, bevor für die Nachricht ein Fehler generiert wird. Ein Zeitablaufwert von 0 Sekunden bedeutet, dass die Steuerung für eine unbestimmte Zeit auf eine Antwort wartet. Die Werte müssen in einem Bereich von 0 bis 255 Sekunden liegen. |
| | Data Table Address (Datentafeladresse) (500CPU- und PLC5-Nachrichtentypen) | Bei einem Lesebefehl ist dies die Adresse im Prozessor, die die Daten zurücksenden soll.<br>Gültige Filetypen: S, B, T, C, R, N und L. |
| | | Bei einem Schreibbefehl ist dies die Adresse im Prozessor, die die Daten empfangen soll.<br>Gültige Filetypen sind I, O, S, B, T, C, R, N, L und RTC(2)(4). |
| | Data Table Offset (Datentafel-Offset) (485CIF-Nachrichtentypen) | Dies ist der Wort-Offset-Wert im allgemeinen Schnittstellenfile (Byte-Offset bei PLC-Geräten) im Zielprozessor, der die Daten sendet. |
| | Local Node Address (Zentrale Netzknotenadresse) | Gibt die Netzknotennummer des Geräts an, das die Nachricht empfängt. Der gültige Bereich liegt für das DH-485-Protokoll zwischen 0 und 31, für das DF1-Protokoll zwischen 0 und 254 bzw. für DeviceNet™ zwischen 0 und 63. |
| | Local/Remote (Zentral/dezentral) | Zentrale oder dezentrale Nachricht. |

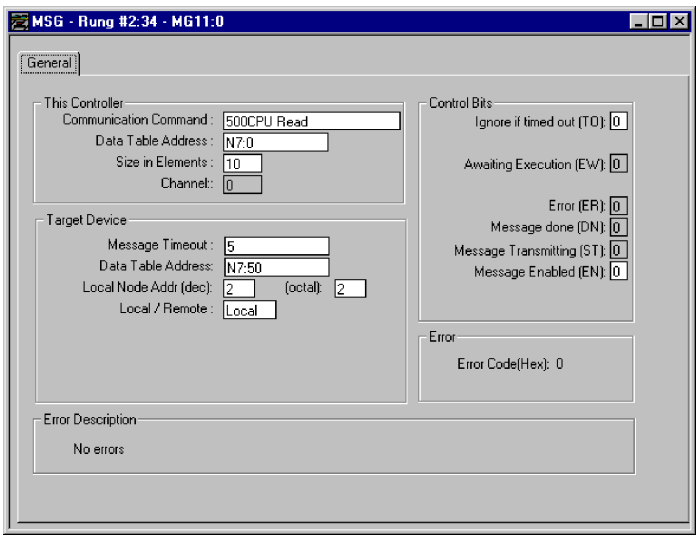
(1) Gilt für MicroLogix 1200 Serie B und höher sowie 1500 Serie B und höher.

(2) 485CIF-Schreibbefehl (nur ST-zu-485CIF).

(3) 500CPU-Schreibbefehl (nur RTC zu Ganzzahl oder RTC zu RTC).

## Beispiel 1 - Zentraler Lesetransfer aus einer 500CPU

*Aufbau des Nachrichtenbefehls*




In diesem Beispiel liest die Steuerung 10 Elemente aus dem File N7 des Zielgeräts (zentraler Netzknoten 2), beginnend mit Wort N7:50. Die 10 Worte werden in den Ganzzahl-File der Steuerung gestellt, beginnend bei Wort N7:0. Wenn bis zum Abschluss der Übertragung mehr als fünf Sekunden vergehen, wird das Fehler-Bit MG11:0/ER als Zeichen für die Zeitüberschreitung durch den Befehl gesetzt.

*Gültige Filetypen-Kombinationen*

Gültige Übertragungen zwischen Filetypen für die MicroLogix-Nachrichtenfunktion sind nachfolgend dargestellt:

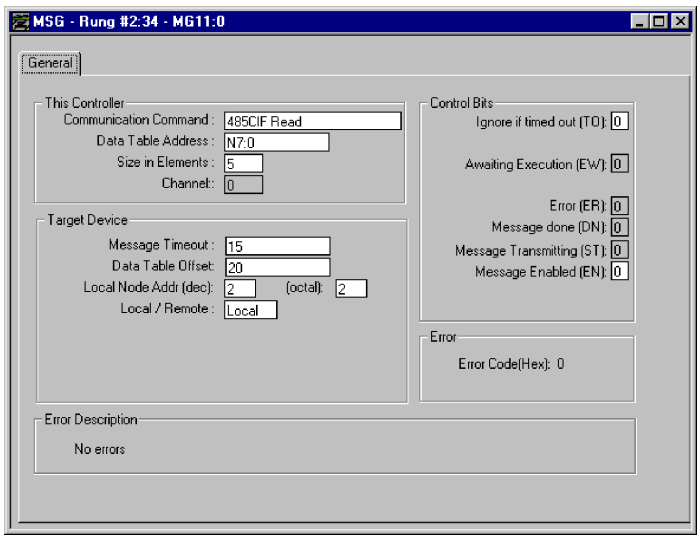
| Zentrale Datentypen | Kommunikationstyp | Zieldatentypen |
|---|---|---|
| O(1), I(1), B, N, L | <---> Lesen/Schreiben | O, I, S, B, N, L |
| T | <---> Lesen/Schreiben | T |
| C | <---> Lesen/Schreiben | C |
| R | <---> Lesen/Schreiben | r |
| RTC(2) | ---> Schreiben | N, RTC |

(1) Ausgangs- und Eingangsdaten sind keine gültigen zentralen Datentypen für Lesenachrichten.

(2) 500CPU-Schreibbefehl (nur RTC zu Ganzzahl oder RTC zu RTC). Gilt nur für MicroLogix 1200 Serie B und höher sowie 1500 Serie B und höher

## Beispiel 2 - Zentraler Lesetransfer aus einem 485CIF

*Aufbau des Nachrichtenbefehls*




In diesem Beispiel liest die Steuerung 5 Elemente (Worte) aus dem CIF-File des Zielgeräts (zentralen Netzknoten 2), beginnend mit Wort 20 (oder Byte 20 bei Nicht-SLC 500-Geräten). Die fünf Elemente werden in den Ganzzahl-File der Steuerung gestellt, beginnend bei Wort N7:0. Wenn bis zum Abschluss der Übertragung mehr als 15 Sekunden vergehen, wird das Fehler-Bit MG11:0/ ER als Zeichen für die Zeitüberschreitung durch den Befehl gesetzt.

*Gültige Filetypen-Kombinationen*

Gültige Übertragungen zwischen Filetypen für die MicroLogix-Nachrichtenfunktion sind nachfolgend dargestellt:

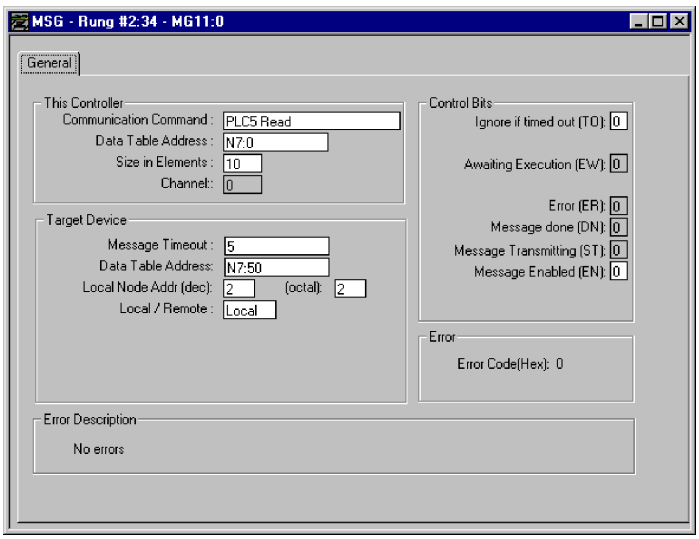
| Zentrale Datentypen | Kommunikationstyp | Zieldatentypen |
|---|---|---|
| O(1), I(1), B, N, L | <---> Lesen/Schreiben | 485CIF |
| T | <---> Lesen/Schreiben | 485CIF |
| C | <---> Lesen/Schreiben | 485CIF |
| R | <---> Lesen/Schreiben | 485CIF |
| ST(2) | ---> Schreiben | 485CIF |

(1) Ausgangs- und Eingangsdaten sind keine gültigen zentralen Datentypen für Lesenachrichten.

(2) Gilt nur für MicroLogix 1200 Serie B und höher sowie 1500 Serie B und höher.

## Beispiel 3 - Zentraler Lesebefehl aus einer PLC-5

*Aufbau des Nachrichtenbefehls*




In diesem Beispiel liest die Steuerung 10 Elemente aus dem File N7 des Zielgeräts (zentraler Netzknoten 2), beginnend mit Wort N7:50. Die 10 Worte werden in den Ganzzahl-File der Steuerung gestellt, beginnend bei Wort N7:0. Wenn bis zum Abschluss der Übertragung mehr als fünf Sekunden vergehen, wird das Fehler-Bit MG11:0/ER als Zeichen für die Zeitüberschreitung durch den Befehl gesetzt.

*Gültige Filetypen-Kombinationen*

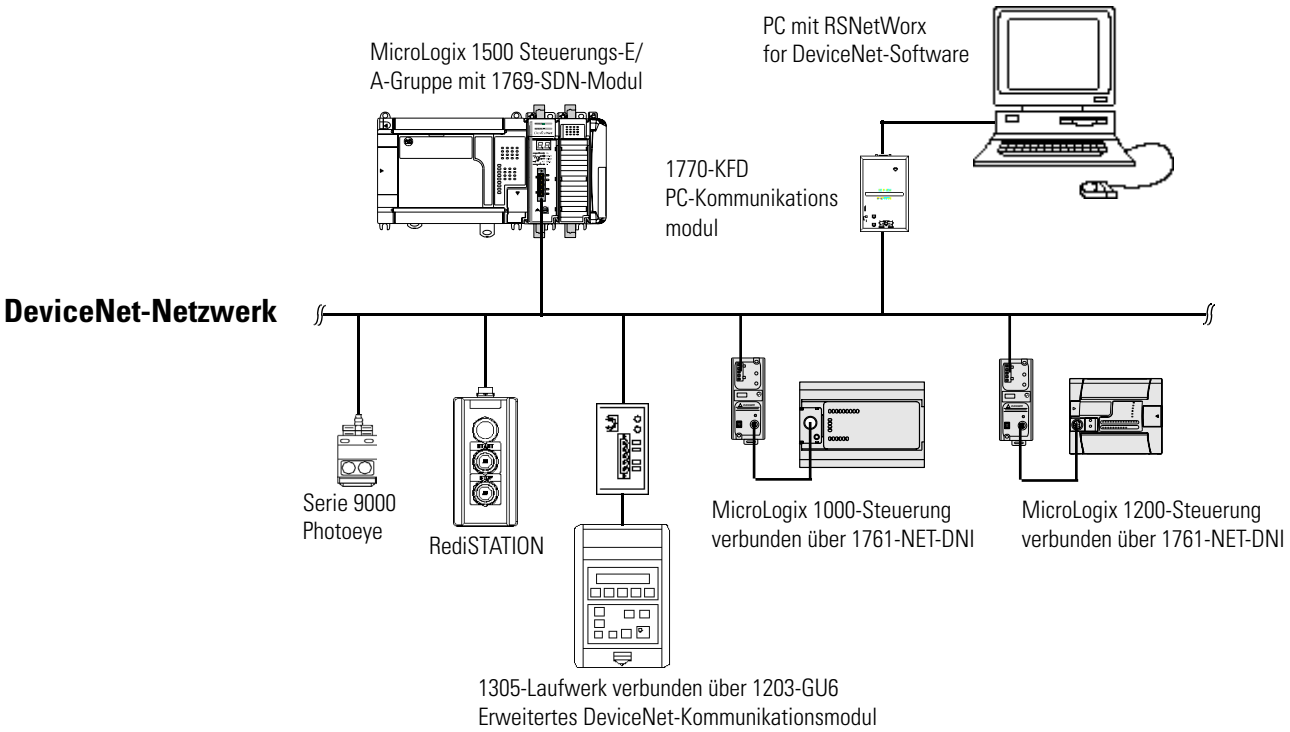
Gültige Übertragungen zwischen Filetypen für die MicroLogix-Nachrichtenfunktion sind nachfolgend dargestellt:

| Zentrale Datentypen | Kommunikationstyp | Zieldatentypen |
|---|---|---|
| O(1), I(1), B, N, L | <---> Lesen/Schreiben | O, I, S, B, N, L |
| T | <---> Lesen/Schreiben | T |
| C | <---> Lesen/Schreiben | C |
| R | <---> Lesen/Schreiben | r |

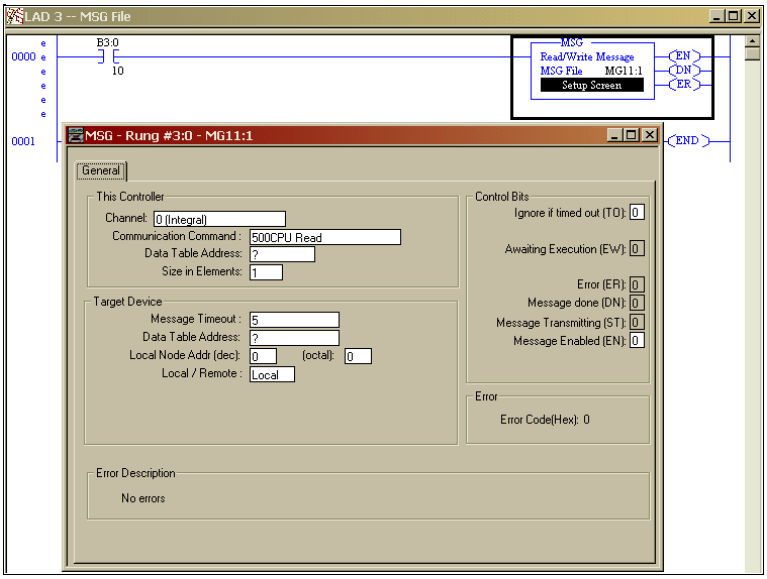
(1) Ausgangs- und Eingangsdaten sind keine gültigen zentralen Datentypen für Lesenachrichten.

## Beispiel 4 – Zentrale DeviceNet-Nachricht konfigurieren

In diesem Abschnitt wird beschrieben, wie Sie eine zentrale Nachricht mithilfe des Scanners und eines MicroLogix 1500 1764-LRP-Prozessors konfigurieren können. Die nachfolgende Abbildung stellt ein Beispiel- netzwerk dar:



*Bildschirm „Nachrichtenkonfiguration“*



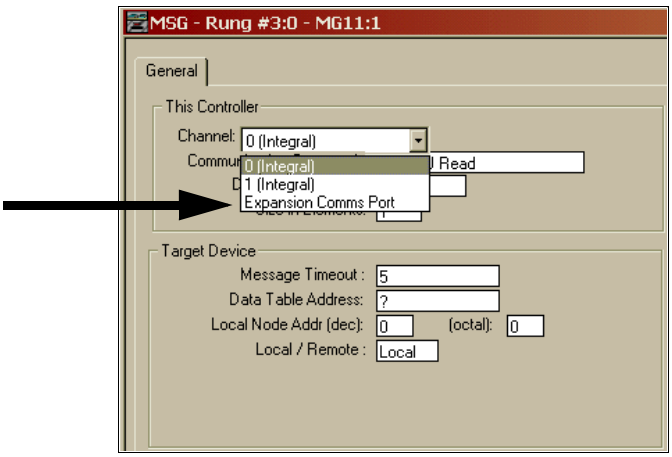

Strompfad 0 zeigt einen standardmäßigen Nachrichtenbefehl (MSG) in RSLogix 500 dar, dem eine Bedingungslogik vorangeht.

1. Klicken Sie doppelt auf den Konfigurationsbildschirm, um den Bildschirm „Message Setup“ (Nachrichtenkonfiguration) zu öffnen.
2. Der RSLogix 500-Bildschirm „Message Setup“ (Nachrichten-konfiguration) wird angezeigt. Über diesen Bildschirm können die Nachrichten-Parameter für die Elemente „This Controller“ (Diese Steuerung), „Target Device“ (Zielgerät) und „Control Bits“ (Steuerbits) konfiguriert und überprüft werden. Im Nachfogenden werden diese Abschnitte näher beschrieben.

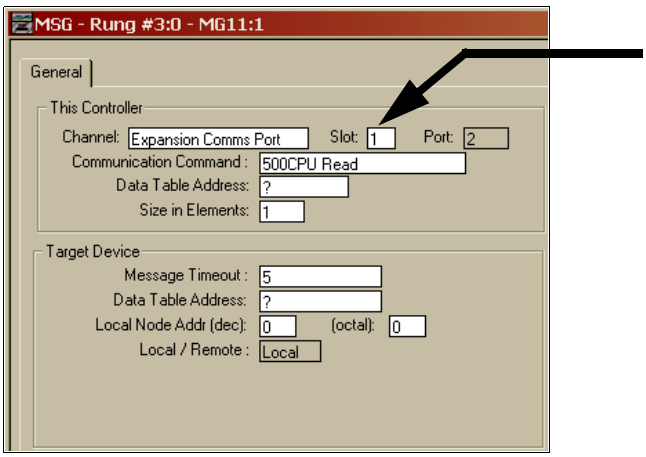
*Parameter für „This Controller„ (diese Steuerung)*

**Kanal**

Der Prozessor 1764-LRP unterstützt drei verschiedene Arten der Nachrichtenübertragung, Kanäle 0 und 1 sind RS-232-Anschlüsse und Kanäle 0 und 1 sind RS-232-Anschlüsse und entsprechen ihrer Funktion nach Kanal 0 der MicroLogix 1200- und MicroLogix 1500-Steuerungen 1764-LSP. Der Prozessor 1764-LRP unterstützt zudem Backplane-Kommunikation durch den Erweiterungs-Kommunikations-anschluss (ECP - Expansion Communication Port), wie im Folgenden dargestellt wird.

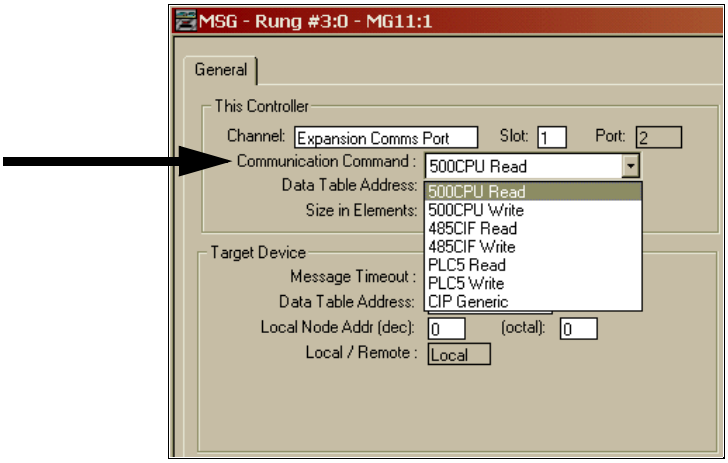



Wenn der ECP (Expansion Communiction Port) ausgewählt wird, können Sie wählen, an welcher Steckplatzposition (1 bis 16) sich der Scanner befinden soll. Der 1764-LRP-Prozessor kann bis zu zwei 1769-SDN-Scannermodule mit voller Nachrichtenfunktion unter- stützen.




**HINWEIS** Sie können mehrere 1769-SDN-Scannermodule in einem 1764-LRP MicroLogix 1500-System verwenden, jedoch nur über die ersten beiden Module Nachrichten versenden. Ein Scanner, der sich physikalisch hinter den ersten beiden befindet, kann nur zum E/A-Scannen verwendet werden.

### Kommunikationsbefehl „CIP Generic“ (CIP-generisch)



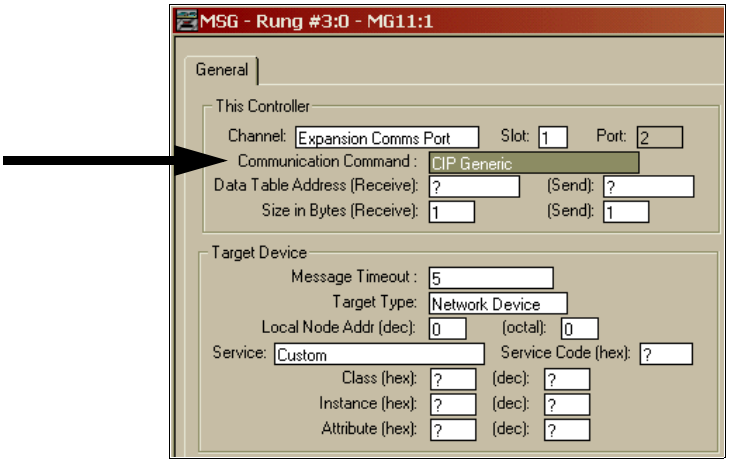

Der 1764-LRP-Prozessor unterstützt die sechs Standardtypen an Kommunikationsbefehlen (ebenso wie alle anderen MicroLogix 1200- und 1500-Steuerungen) und „CIP-generisch“ am Erweiterungs-Kommunikationsanschluss. Wenn einer der sechs Standardbefehle gewählt wurde, können Sie eine Standardnachricht zu einem mit DeviceNet-Produkten verbundenen Zielgerät initiieren. Die DeviceNet-Produkte müssen PCCC-Nachrichten unterstützen, dazu gehören z. B. MicroLogix- und SLC-Steuerungen mit 1761-NET-DNIs, 1203-GU6-Laufwerk und andere MicroLogix 1500-Steuerungen mit 1769-SDN-Scannermodulen. Über DeviceNet können Sie das Lesen, Schreiben, Hoch-/Herunterladen von Programmen und die Online-Überprüfung initiieren. Diese Funktionsweise entspricht der Arbeits- weise in DH-485- und DH+-Netzwerken.

CIP steht für „Control & Information Protocol“ (Steuer- und Informationsprotokoll). CIP ist ein neueres und vielfältigeres Protokoll als PCCC. Es ist ein offenes Protokoll, das von neueren Allen-Bradley-Steuerungen und Produkten von Drittanbietern unterstützt wird.

Bei CIP-Nachrichten handelt es sich um das native Nachrichtenformat für DeviceNet. Alle DeviceNet-Geräte sind mit der CIP-Nachrichten- funktion kompatibel. Der MicroLogix 1500 1764-LRP-Prozessor (Serie C) verfügt über einen erweiterten Nachrichtenbefehl, mit dem sich mühelos CIP-Nachrichten versenden lassen.

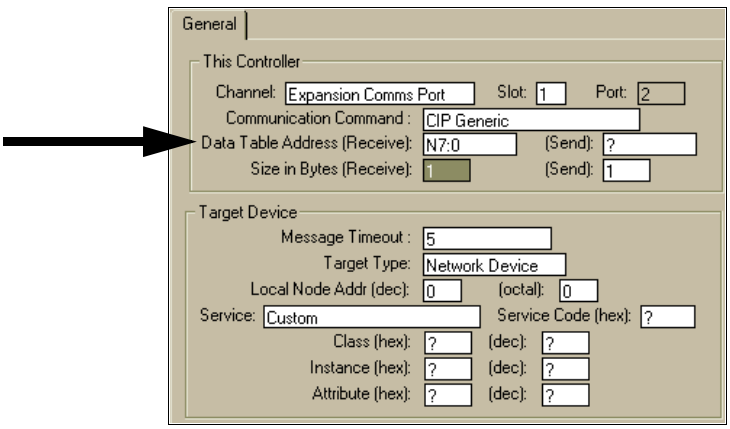
Wenn Sie „CIP-generisch“ wählen, wird der Nachrichtenbefehl so konfiguriert, dass er mit DeviceNet-Geräten kommuniziert, die keine PCCC-Nachrichten unterstützen. Wenn „CIP-generisch“ gewählt wurde,

ändern sich eine Reihe an Nachrichten-Parametern und viele neue Parameter werden abhängig vom ausgewählten Dienst aktiviert.




**Data Table Adress (Receive and Send) (Datentafeladresse (empfangen und senden))**

Dieser Wert bezeichnet den Datenfilestandort innerhalb der 1764-LRP-Steuerung, die Daten vom DeviceNet-Gerät empfängt, und/oder die Ausgangsadresse des Datenfiles, die zum DeviceNet-Zielgerät gesendet wird.



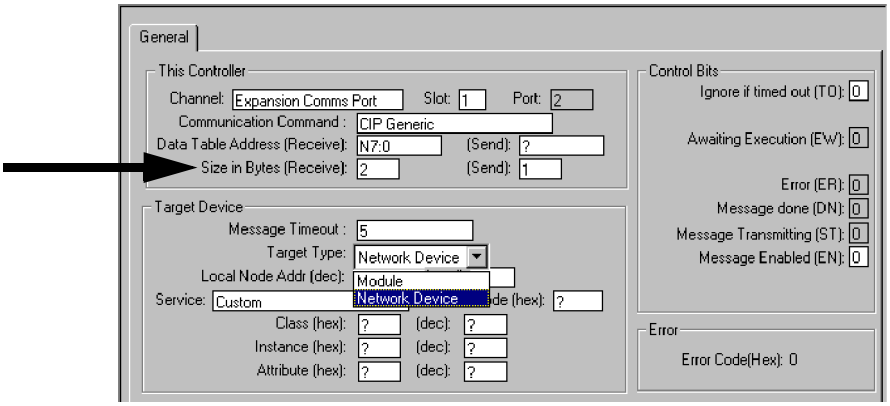

**Size in Bytes (Receive and Send) (Größe in Byte (empfangen und senden))**

Da alle Daten, die über DeviceNet übermittelt werden, auf Byte basieren, müssen Sie die Anzahl der gesendeten und empfangenen Byte eingeben. Vergewissern Sie sich, dass genügend Speicher für das Zielgerät vorhanden ist. Wortelemente in 1764-LRP-Steuerungen enthalten alle 2 Byte. Dies gilt für Bit- und Ganzzahl-Datenfiles. Doppelwort- und Fließkommaelemente enthalten jeweils 4 Byte.

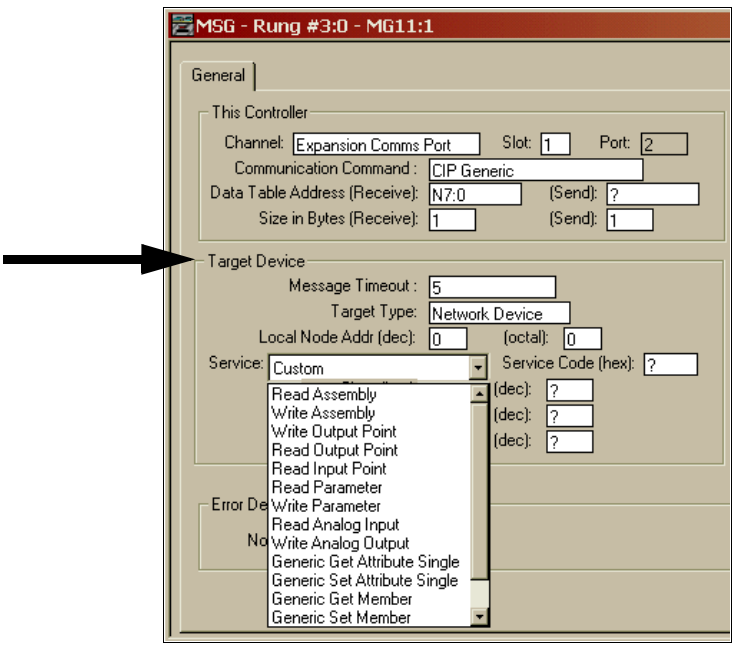
Zum Empfangen muss die eingegebene Größe in Byte größer als oder gleich der Anzahl der Byte sein, die das DeviceNet-Gerät zurückgibt. DeviceNet-Geräte geben abhängig von Klasse und Dienst eine festgelegte Anzahl an Byte zurück. Wenn mehr Daten als erwartet zurückgegeben werden, tritt in der Nachricht ein Fehler auf und es werden keine Daten geschrieben.

Wenn weniger Daten als erwartet zurückgegeben werden, werden die Daten geschrieben und die übrigen Byte mit Null aufgefüllt.

Im nachfolgenden Beispiel erhält N7:0 2 Byte (1 Wort) an Daten.




*Zielgerät*




**Message Timeout (Nachrichten-Zeitablauf)**

Der Nachrichten-Zeitablauf wird in Sekunden angegeben. Wenn das Ziel nicht innerhalb dieser Zeit reagiert, wird durch den Nachrichten- befehl ein bestimmter Fehler generiert (siehe „Fehlercodes zu MSG-Befehlen“ auf Seite 21-43). Die annehmbare Zeitspanne sollte nach den Erfordernissen der Anwendung und der Kapazität bzw. Auslastung des Netzwerks bemessen sein.

### Target Type (Zieltyp)

SIE können entweder „Module“ (Modul) oder „Network Device“ (Netzwerkgerät) wählen. Wenn Sie eine Nachricht an ein Gerät über DeviceNet senden möchten, wählen Sie die Option für das Netzwerk- gerät. Wenn Sie eine Nachricht an einen DeviceNet-Paramter des Scanners senden möchten, wählen Sie die Option für das Modul. Dadurch kann das Steuerungsprogramm auf die Modulparameter zugreifen.

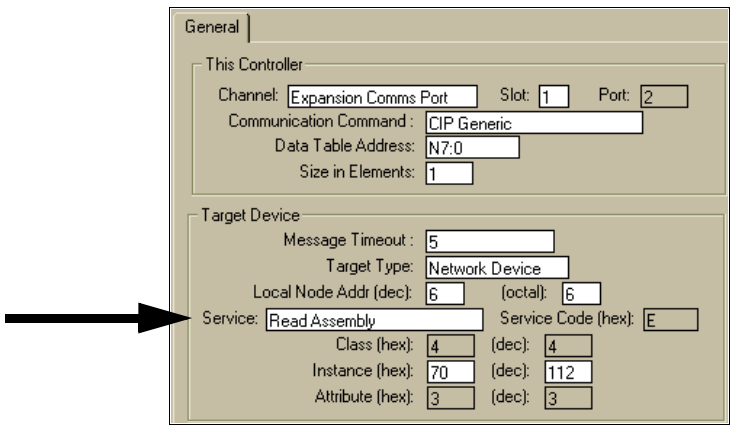
**HINWEIS** Beachten Sie, dass Sie viele Modulparameter nicht bearbeiten können und dass einige Parameter nur dann bearbeitet werden können, wenn sich das Modul im Leerlauf befindet.

### Local Node Address (Zentrale Netzknotenadresse)

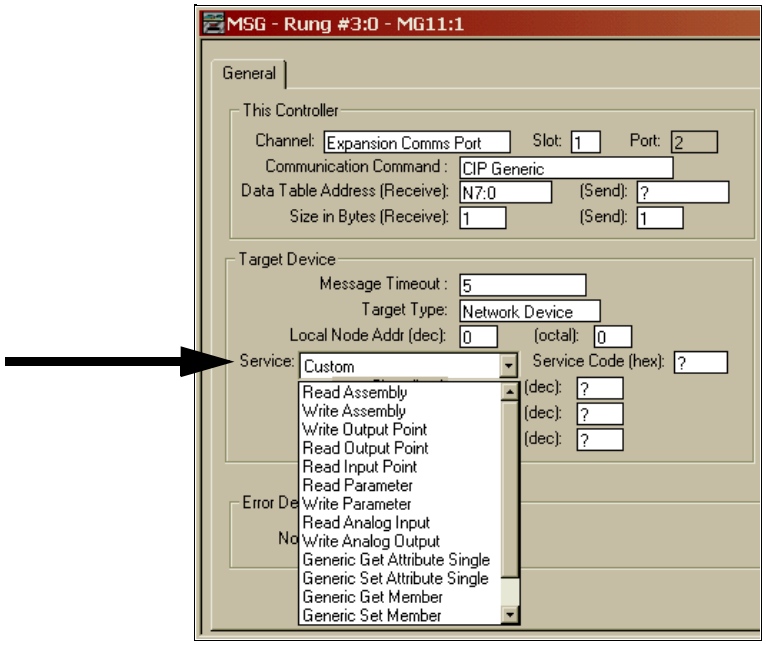
Dies ist die DeviceNet-Netzknotennummer des Zielgeräts.

### Service (Dienst)

DeviceNet verwendet Dienste, um bestimmte Nachrichtenfunktionen auszuführen. Viele der standardmäßigen Dienste und die entsprechen- den Parameter wurden bereits so konfiguriert, dass sie sofort verwen- det werden können.



Wenn Sie einen Dienst verwenden möchten, der nicht verfügbar ist, wählen Sie eine der generischen Dienste. Mithilfe des generische Dienstes können Sie bestimmte Dienstcode-Parameter eingeben. Informationen darüber, welche Dienste von einem Zielgerät unter- stützt werden, erhalten Sie in der Regel in der Dokumentation des Gerätes.



# Dezentrale Nachrichten

Über die Steuerung ist auch eine dezentrale Nachrichtenübertragung möglich. Dabei handelt es sich um die Fähigkeit zum Austausch von Informationen mit einem Gerät, das nicht mit dem zentralen Netzwerk verbunden ist. Bei dieser Verbindungsart ist auf dem zentralen Netz- werk ein Gerät erforderlich, das die Aufgabe einer Brücke oder eines Gateway zum anderen Netzwerk hat.

## Dezentrale Netzwerke

*DH-485- und DH+-Netzwerke*

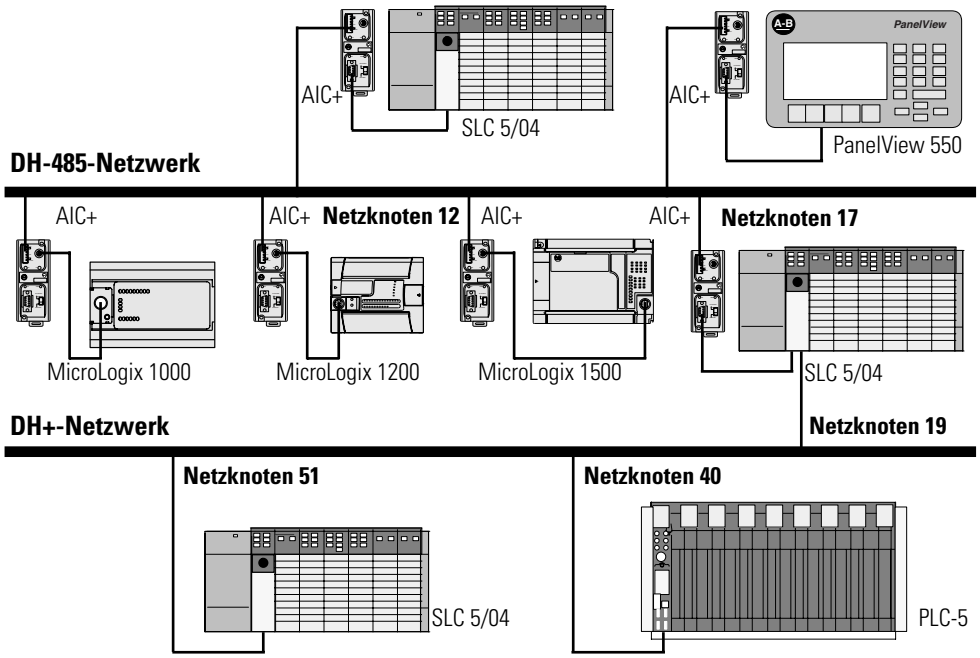
Die nachfolgende Abbildung zeigt zwei Netzwerke: ein DH-485- und ein DH+-Netzwerk. Der SLC 5/04-Prozessor am DH-485-Netzknoten 17 wurde für eine Passthru-Operation konfiguriert. Geräte, die in der Lage sind, dezentrale Nachrichten zu empfangen, und die an eines dieser Netzwerke angeschlossen sind, können entsprechend der Leistungsmerkmale des jeweiligen Geräts Lese- oder Schreibbefehle mit anderen Geräten in dem anderen Netzwerk austauschen. In diesem Beispiel ist Netzknoten 12 auf DH-485 eine MicroLogix 1500-
Steuerung. Die MicroLogix 1500-Steuerung kann auf dezentrale Nachrichtenaufforderungen von den Netzknoten 40 oder 51 auf dem DH+-Netzwerk reagieren und eine Nachricht für einen beliebigen Knoten im DH+-Netzwerk initiieren.

**HINWEIS** Die MicroLogix 1000-Steuerung kann auf dezentrale Nachrichtenaufforderungen reagieren, aber keine derartigen Nachrichtenübertragungen starten.

**HINWEIS** In diesem Beispiel weisen die MicroLogix 1200-Steuerungen und die MicroLogix 1500-Steuerungen dieselben Funktionen auf.

Diese Funktionalität steht auch auf Ethernet-Netzwerken zur Ver- fügung, wenn SLC 5/04 an DH-485-Netzknoten 17 durch einen SLC 5/05-Prozessor ersetzt wird.

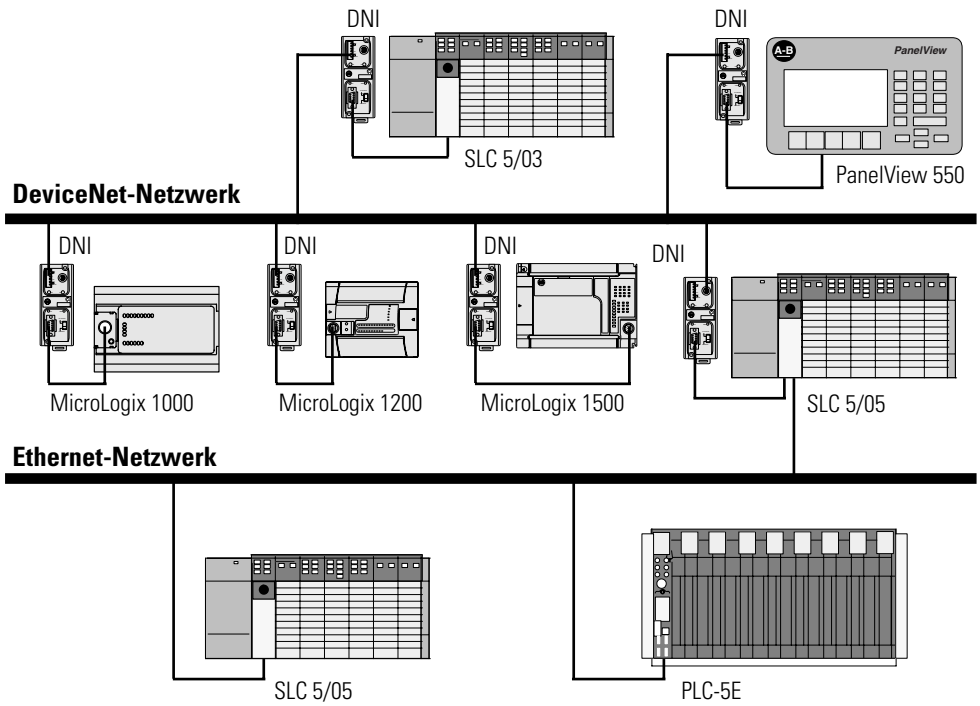
**Abbildung 21.1 DH-485- und DH+-Netzwerke**




*DeviceNet- und Ethernet-Netzwerk*

Die nachfolgende Abbildung zeigt ein DeviceNet-Netzwerk mit DeviceNet-Schnittstellen (1761-NET-DNI), das an ein Ethernet-Netzwerk mit einem SLC 5/05-Prozessor angeschlossen ist. In dieser Konfiguration können Steuerungen in dem DeviceNet-Netzwerk auf Aufforderungen von Geräten in dem Ethernet-Netzwerk reagieren, aber keine Kommunikation mit diesen Geräten initiieren.

**Abbildung 21.2 DeviceNet- und Ethernet-Netzwerk**



# Dezentrale Nachrichtenübertragung konfigurieren

Sie führen die Konfiguration für die dezentrale Fähigkeit über den RSLogix 500-Bildschirm zur Nachrichtenkonfiguration durch.

## Konfigurationsbildschirm und Netzwerk - Beispiel

Die unten dargestellte Nachrichtenkonfiguration gilt für MicroLogix 1500 an Netzknoten 12 im DH-485-Netzwerk. Diese Nachricht liest fünf Datenelemente der SLC 5/04 (Netzknoten 51 im DH+-Netzwerk), beginnend bei Adresse N:50:0. Die SLC 5/04 an Netzknoten 23 des DH+-Netzwerks ist für den Passthru-Betrieb konfiguriert.

**HINWEIS** In diesem Beispiel weisen die MicroLogix 1200-Steuerungen und die MicroLogix 1500-Steuerungen dieselben Funktionen auf.

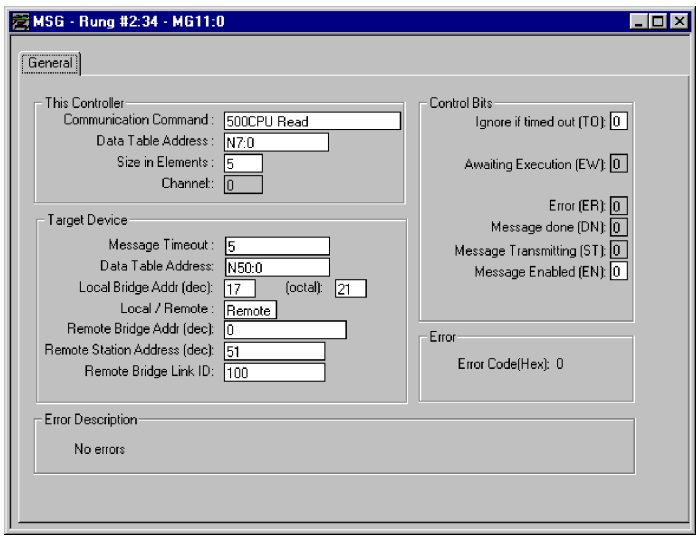
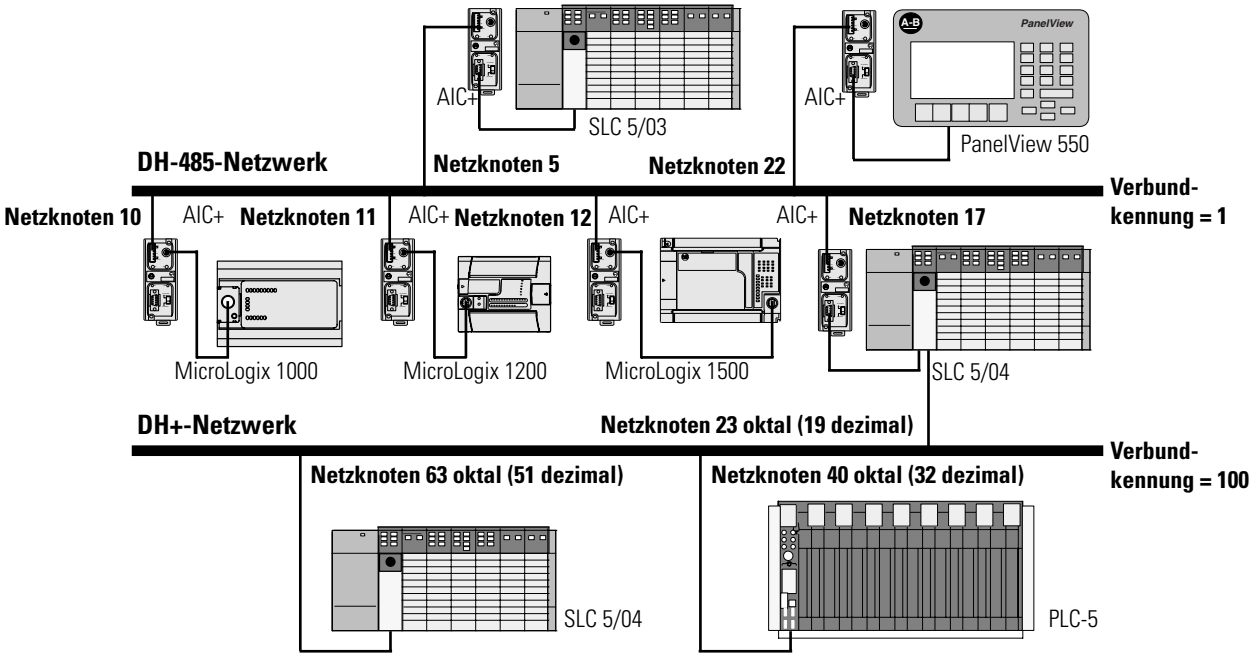


**Abbildung 21.3 DH-485- und DH+-Beispielnetzwerk**

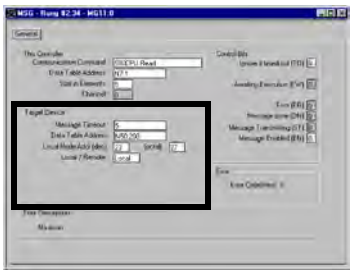



## Parameter für „This Controller" (diese Steuerung)

Siehe „Parameter für „Target Device" (Zielgerät)" auf Seite 21-23.

## Parameter für „Control Bits" (Steuer-Bits)

Siehe „„Control Bits"- Parameter (Steuer-Bits-Parameter)" auf Seite 21-10.



## Parameter für „Target Device" (Zielgerät)

*Nachrichten-Zeitablauf*

Siehe „Nachrichten-Zeitablauf" auf Seite 21-23.

*Datentafeladresse*

Siehe „Datentafeladresse/-Offset" auf Seite 21-24.

*Zentrale Brückenadresse*

Diese Variable bezeichnet die Brückenadresse in dem zentralen Netzwerk. In diesem Beispiel schreibt DH-485-Netzknoten 12 (MicroLogix 1500 auf

Verbundkennung 1) Daten auf Netzknoten 51 (SLC 5/04 auf Verbundkennung 100). SLC 5/04 an Netzknoten 17 ist das Brückengerät.

Diese Variable sendet die Nachricht an den lokalen Netzknoten 17.

### *Dezentrale Brückenadresse*

Diese Variable bezeichnet die dezentrale Netzknotenadresse des Brückengeräts. In dem dargestellten Beispiel ist die dezentrale Brückenadresse auf null gesetzt, weil das Zielgerät, SLC 5/04 an Netzknoten 63 (oktal) die dezentrale Nachrichtenübertragung unter- stützt. Wenn das Zielgerät die dezentrale Nachrichtenübertragung unterstützt, ist keine dezentrale Brückenadresse erforderlich. Wenn das Zielgerät die dezentrale Nachrichtenübertragung nicht unterstützt (SLC 500, SLC 5/01, SLC 5/02 und MicroLogix 1000 Serien A, B und C), ist die dezentrale Brückenadresse erforderlich.

### *Dezentrale Stationsadresse*

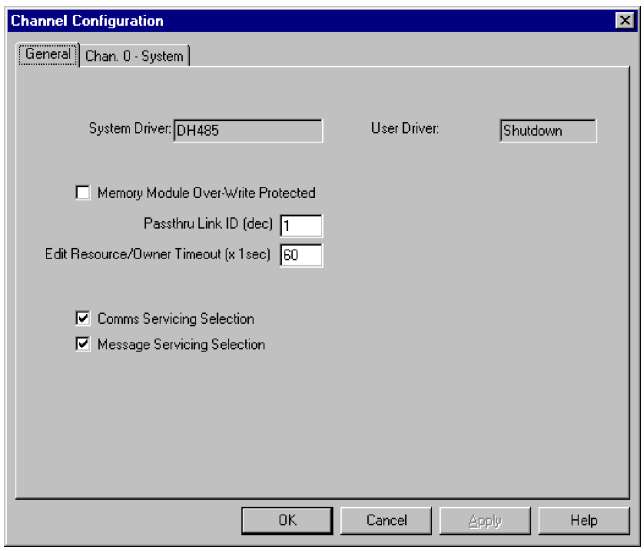
Diese Variable ist die endgültige Zieladresse des Nachrichtenbefehls. In dem dargestellten Beispiel empfangen die Elemente 0 bis 4 der Verbundkennung 100 der SLC 5/04 an Netzknoten 63 (oktal) Daten von der MicroLogix 1500-Steuerung an Netzknoten 12 unter der Verbundkennung 100.

### *Verbundkennung der dezentralen Brücken*

Diese Variable ist ein benutzerdefinierter Wert, der das dezentrale Netzwerk als Nummer darstellt. Diese Nummer muss von jedem Gerät verwendet werden, das eine dezentrale Nachrichtenübertragung zu diesem Netzwerk initiiert. In dem Beispiel muss jede Steuerung unter der Verbundkennung 100, die Daten an ein Gerät unter der Verbundkennung 100 sendet, die dezentrale Brückenverknüpfungsnummer des Passthru-
Geräts verwenden. In diesem Beispiel ist die SLC 5/04 unter Verbundken- nung 1, Netzknoten 17, das Passthru-Gerät.

### *Passthru-Verbundkennung*

Geben Sie in dem Kanal-Konfigurationsbildschirm die Passthru-Verbundkennung in dem Register „Allgemein“ ein. Die Verbundken- nung ist ein benutzerdefinierter Wert zwischen 1 und 65 535. Alle Geräte, die die dezentrale Nachrichtenübertragung initiieren und mit dem zentralen Netzwerk verbunden sind, müssen für diese Variable denselben Wert verwenden.



## Fehlercodes zu MSG-Befehlen

Wenn der Prozessor während der Übertragung von Nachrichtendaten einen Fehler erkennt, wird das ER-Bit gesetzt und ein Fehlercode eingegeben, der über die Programmiersoftware überwacht werden kann.

| Fehlercode | Beschreibung des Fehlers |
|---|---|
| 02H | Zielnetzknoten ist besetzt. Keine Bestätigung. Max. Anzahl der Wiederholungen durchgeführt. |
| 03H | Zielnetzknoten antwortet nicht, da Nachricht zu groß ist. |
| 04H | Zielnetzknoten kann nicht antworten, da er die Befehlsparameter nicht erkennt ODER da der Steuerblock möglicherweise unbeabsichtigt geändert wurde. |
| 05H | Der zentrale Prozessor ist offline (möglicherweise doppelter Netzknoten vorhanden). |
| 06H | Zielnetzknoten kann nicht antworten, da die angeforderte Funktion nicht unterstützt wird. |
| 07H | Zielnetzknoten antwortet nicht. |
| 08H | Zielnetzknoten kann nicht antworten. |
| 09H | Zentrale Modemverbindung wurde unterbrochen. |
| 0BH | Zielnetzknoten akzeptiert diese Art von MSGBefehl nicht. |
| 0CH | Rücksetzung des MasterVerbunds (eine mögliche Quelle ist der DF1Master). |
| 10H | Zielnetzknoten kann nicht antworten, da ein falscher Befehlsparameter oder ein nicht unterstützter Befehl eingegeben wurde. |
| 12H | KonfigurationsProtokollfehler im zentralen Kanal vorhanden. |
| 13H | Zentraler MSGKonfigurationsfehler in den dezentralen MSGParametern. |
| 15H | Konfigurationsparameterfehler im zentralen Kanal. |
| 16H | Ziel oder zentrale Brückenadresse ist höher als größte NetzknotenAdresse. |
| 17H | Zentrale Bearbeitung nicht unterstützt. |
| 18H | Rundsenden nicht unterstützt. |
| 21H | Falscher MSG-Fileparameter zum Aufbau der Nachricht. |
| 30H | PCCC-Beschreibung: Dezentraler Stations-Host nicht vorhanden, nicht angeschlossen oder ausgeschaltet. |
| 37H | Zeitablauffehler der Nachricht im zentralen Prozessor. |
| 39H | Neukonfiguration des zentralen Kommunikationskanals bei aktivem MSG. |
| 3AH | STS in Antwort von Ziel ungültig. |
| 40H | PCCC-Beschreibung: Host konnte Aufgabe aufgrund von Hardware-Fehler nicht durchführen. |
| 45H | MSG-Antwort kann nicht verarbeitet werden. Daten in MSG-Leseantwort unzureichend oder Netzwerkadressenparameter ungültig. |
| 50H | Speicherkapazität des Zielnetzknotens erschöpft. |
| 60H | Zielnetzknoten kann nicht antworten, da File geschützt ist. |
| 70H | PCCC-Beschreibung: Prozessor im Programm-Modus. |
| 80H | PCCC-Beschreibung: Kompatibilitätsmodusfile nicht vorhanden oder Kommunikationszonenproblem. |
| 90H | PCCC-Beschreibung: Dezentrale Station kann Befehl nicht in Puffer stellen. |
| B0H | PCCC-Beschreibung: Problem an dezentraler Station wegen Herunterladevorgang. |
| C0H | PCCC-Beschreibung: Befehl kann wegen aktiver IPBs nicht ausgeführt werden. |
| D0H | Eine der folgenden Ursachen:<br>• Keine IP-Adresse für das Netzwerk konfiguriert.<br>• Ungültiger Fehler - freilaufender Nachrichtenfehler.<br>• Ungültige Adresse - freilaufender Nachrichtenfehler.<br>• Kein Privileg - freilaufender Nachrichtenfehler. |
| D1H | Maximale Verbindungen belegt - keine Verbindung verfügbar |
| D2H | Internet-Adresse oder Host-Name ungültig |
| D3H | Host nicht vorhanden/Host kann nicht mit Namens-Server kommunizieren |
| D4H | Verbindung nicht hergestellt innerhalb von benutzerdefiniertem Zeitablauf |
| D5H | Verbindungszeitablauf durch Netzwerk |
| D7H | Verbindung von Ziel-Host abgelehnt |
| D8H | Verbindung unterbrochen |

| Fehlercode | Beschreibung des Fehlers |
|---|---|
| D9H | Antwort nicht empfangen innerhalb von benutzerdefiniertem Zeitablauf |
| DAH | Kein Netzwerkpuffer verfügbar |
| E1H | PCCC-Beschreibung: Ungültiges Adressformat, ungültiger Wert in einem Feld |
| E2H | PCCC-Beschreibung: Ungültiges Adressformat, nicht genügend Felder angegeben |
| E3H | PCCC-Beschreibung: Ungültiges Adressformat, zu viele Felder angegeben |
| E4H | PCCC-Beschreibung: Ungültige Adresse, Symbol nicht vorhanden |
| E5H | PCCC-Beschreibung: Ungültiges Adressformat, Symbol gleich 0 oder größer als maximale Zeichenzahl für dieses Gerät |
| E6H | PCCC-Beschreibung: Ungültige Adresse, Adresse nicht vorhanden oder verweist nicht auf verwendbare Daten |
| E7H | Zielnetzknoten kann nicht antworten, da eine zu große Länge angefordert wurde. |
| E8H | PCCC-Beschreibung: Aufforderung kann nicht durchgeführt werden, Situationsänderung (z. B. Filegröße) während Multipaket–Vorgang |
| E9H | PCCC-Beschreibung: Daten oder File zu groß. Kein Speicherplatz verfügbar. |
| EAH | PCCC-Beschreibung: Anforderung zu lang; Transaktionsgröße plus Wortadresse ist zu lang. |
| EBH | Zielnetzknoten kann nicht antworten, da Zielnetzknoten Zugriff verweigert. |
| ECH | Zielnetzknoten kann nicht antworten, da die angeforderte Funktion derzeit nicht verfügbar ist. |
| EDH | PCCC-Beschreibung: Ressource ist bereits verfügbar; Zustand bereits vorhanden. |
| EEH | PCCC-Beschreibung: Befehl kann nicht ausgeführt werden |
| EFH | PCCC-Beschreibung: Überlauf; Histogrammüberlauf. |
| F0H | PCCC-Beschreibung: Kein Zugriff |
| F1H | Zentraler Prozessor stellt ungültigen Zielfiletyp fest. |
| F2H | PCCC-Beschreibung: Unzulässiger Parameter; unzulässige Daten im Such- oder Befehlsblock. |
| F3H | PCCC-Beschreibung: Adresse verweist auf gelöschten Bereich |
| F4H | PCCC-Beschreibung: Fehler bei Befehlsausführung aus unbekanntem Grund; PLC-3-Historgrammüberlauf. |
| F5H | PCCC-Beschreibung: Datenkonvertierungsfehler |
| F6H | PCCC-Beschreibung: Der Scanner kann nicht mit einem Rackadapter 1771 kommunizieren. Dies könnte darauf zurückzuführen sein, dass der Scanner keine Abfrage durchführt, der ausgewählte Adapter nicht abgefragt wird, der Adapter nicht antwortet oder eine ungültige DCM-BT-Aufforderung (Blocktransfer) vorliegt. |
| F7H | PCCC-Beschreibung: Der Adapter kann nicht mit einem Modul kommunizieren. |
| F8H | PCCC-Beschreibung: Die Antwort des Moduls 1771 ist nicht gültig - Größe, Prüfsumme usw. |
| F9H | PCCC-Beschreibung: Bezeichnung doppelt verwendet |
| FAH | Zielnetzknoten kann nicht antworten, da ein anderer Netzknoten die ausschließliche Zugriffsberechtigung auf den File besitzt. |
| FBH | Zielnetzknoten kann nicht antworten, da ein anderer Netzknoten der ausschließliche Programmbenutzer ist (die alleinige Zugriffsberechtigung auf alle Files besitzt). |
| FCH | PCCC-Beschreibung: Diskettenfile ist schreibgeschützt oder Zugriff nicht möglich (nur offline). |
| FDH | PCCC-Beschreibung: Diskettenfile wird von einer andere Anwendung verwendet; Aktualisierung nicht durchgeführt (nur offline). |
| FFH | Zentraler Kommunikationskanal ist ausgeschaltet. |

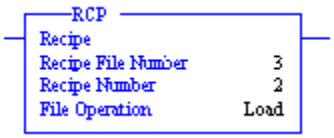
**HINWEIS** Für Benutzer des Referenzhandbuchs zum 1770-6.5.16 DF1-Protokoll und Befehlssatz: Der MSG-Fehlercode spiegelt das STS-Feld der Antwort auf den MSG-Befehl wider.

- Die Codes E0 bis EF repräsentieren die EXT STS-Codes 0 bis F.
- Die Codes F0 bis FC repräsentieren die EXT STS-Codes 10 bis 1C.

# Rezept (nur MicroLogix 1500) und Datenprotokollierung (nur MicroLogix 1500 1764-LRP-Prozessoren)

Dieses Kapitel erläutert, wie Rezept- und Datenprotokollierungs- funktionen verwendet werden.

## RCP – Rezept (nur MicroLogix 1500)




Befehlstyp: Ausgang

**Ausführungszeit des RCP-Befehls**

| Steuerung | Operation | Strompfad | |
|---|---|---|---|
| | | wahr | unwahr |
| MicroLogix 1500 | Laden | 30,7 µs + 7,9 µs/Wort<br>+ 13,8 µs/Doppelwort oder Fließkomma | 0,0 µs |
| | Speichern | 28,5 µs + 8,5 µs/Wort<br>+ 15,1 µs/Doppelwort oder Fließkomma | 0,0 µs |

Mithilfe des RCP-Files können Sie benutzerdefinierte Datenlisten speichern, die mit einem Rezept in Zusammenhang stehen. Wenn Sie diese Files und den RCP-Befehl verwenden, können Sie Datensätze zwischen der Rezept-Datenbank und einem Satz an benutzer- spezifischen Standorten im Steuerungsfile-System übertragen.

Wenn Sie einen Rezeptfile erstellen, können Sie wählen, ob die Rezeptdaten im Speicher des Benutzerprogramms oder der Datenprotokoll-Warteschlange gespeichert werden sollen.

**WICHTIG** Die Option für die Datenprotokoll-Warteschlange kann nur mit 1764-LRP MicroLogix 1500-Steuerungen der Serie C oder höher verwendet werden. Wenn Sie eine 1764-LSP MicroLogix 1500-Steuerung verwenden, müssen Sie die Option für das Benutzerprogramm wählen.

Dieser Abschnitt behandelt folgende Themen:


- „Rezeptfile und Programmierbeispiel“ auf Seite 22-3
- „Beispielwarteschlange 0“ auf Seite 22-8
- „Beispielwarteschlange 5“ auf Seite 22-9
- „Lade-Tools“ auf Seite 22-16
- „Informationen zum Erstellen Ihrer eigenen Anwendung“ auf Seite 22-17

Die folgenden Punkte können Ihnen bei der Wahl des Speichertyps behilflich sein:

- Wenn Sie den Benutzerprogramm-Speicher verwenden, haben Sie den Vorteil, dass Rezeptdaten im Speichermodul der Steuerung gespeichert werden. Wenn die Datenprotokoll-Warteschlange verwendet wird, können die Rezeptdaten nicht im Speichermodul der Steuerung gespeichert werden.
- Wenn Sie den Speicher der Datenprotokoll-Warteschlange verwen- den, haben Sie den Vorteil, dass die Rezeptdaten keinen Speicher- platz im Benutzerprogramm belegen. Wenn Sie die Datenproto- koll-Funktion nicht verwenden und den Speicher der Datenproto- koll-Warteschlange wählen, steht Ihnen zusätzlicher Speicherplatz von bis zu 48 KB für RCP-Files zur Verfügung. *Sie können die Datenprotokoll-Warteschlange für Datenprotokolle und Rezeptdaten verwenden, die Gesamtgröße darf jedoch 48 KB nicht überschreiten.*
- Wenn Sie die Datenprotokoll-Warteschlange für einen RCP-File verwenden, benutzen alle RCP-Files in Ihrem Projekt den Spei- cherplatz der Datenprotokoll-Warteschlange.

Nähere Informationen zum Rezeptfile-Verfahren finden Sie in Schritt 2, „Erstellen Sie einen RCP-File“, auf Seite 22-3.

Der RCP-Befehl verwendet die folgenden Parameter:

- Rezeptfilenummer – Mit dieser Filenummer wird die benutzerdefinierte Liste mit Adressen identifiziert, die mit dem Rezept in Zusammenhang stehen.
- Rezeptnummer – Bestimmt die Nummer des zu verwendenden Rezepts. Wenn die Rezeptnummer ungültig ist, tritt ein Benutzer- fehler (Code 0042) auf.
- File-Operation – Bestimmt, ob es sich bei der Operation um ein Laden aus der Datenbank oder ein Speichern in der Datenbank handelt.

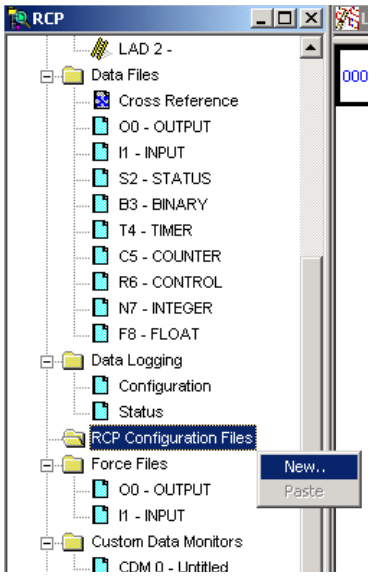
Wenn der RCP-Befehl auf einen wahren Strompfad angewendet wird, werden Daten zwischen der Rezeptdatenbank und den bestimmten Datenstandorten übertragen.

In der folgenden Tabelle werden Adressierungsmodi und Filetypen dargestellt:

**Tabelle 22.1 Gültige Adressierungsmodi und Filetypen für RCP-Befehl**

*Definitionen der in dieser Tabelle verwendeten Begriffe finden Sie im Abschnitt „Befehlsbeschreibungen verwenden“ auf Seite 4-2.*

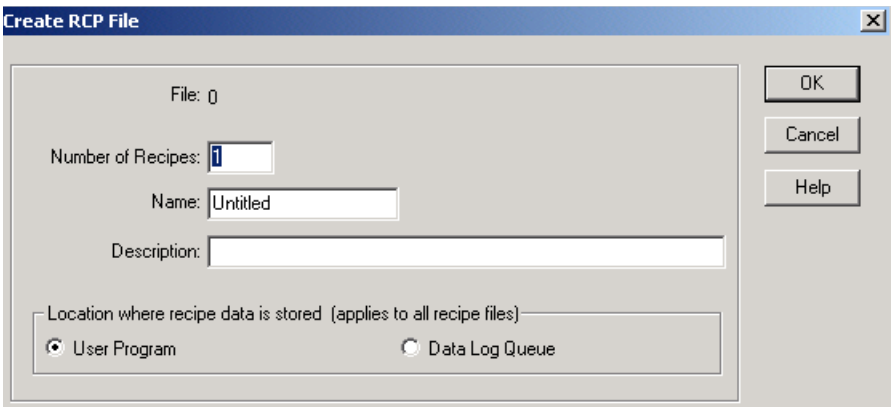
| Parameter | Datenfiles | | | | | | | | | | Funktionsfiles | | | | | | | | | Endschalter | | | Adressierungsmodus | | | Adressierungsebene | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | O | I | S | B | T, C, R | N | F | ST | L | MG, PD | RTC | HSC | PTO, PWM | STI | EII | BHI | MMI | DAT | TPI | PLS - Programmierb. | CS - Komm | IOS - E/A | unmittelbar | direkt | indirekt | Bit | Wort | Doppelwort | Fließkomma | Element |
| Rezeptnummer | | | | | | | | | | | | | | | | | | | | | | | • | | | | | | | |
| File | • | • | | • | | • | | | | | | | | | | | | | | | | | • | • | | | • | | | |




## Rezeptfile und Programmierbeispiel

### *Konfigurieren des RCP-Files*

1. Suchen und wählen Sie in RSLogix 500 die *RCP Configuration Files* (RCP-Konfigurationsfiles). Klicken Sie mit der rechten Maustaste auf die Files und wählen Sie aus dem Menü die Option *New* (Neu).
2. Erstellen Sie einen RCP-File.




- File – Dies ist die Nummer, die den RCP-File kennzeichnet. Es ist die *Rezeptfilenummer*, die im RCP-Befehl in dem Kontaktplan- programm verwendet wird und die Rezept-Datenbank kennzeichnet.
- Number of Recipes (Rezeptanzahl) – Dies ist die Anzahl an Rezepten, die im RCP-File enthalten sind. Es können nie mehr als 256 sein. Dies ist die *Rezeptnummer*, die im RCP-Befehl im Kontaktplanprogramm verwendet wird.
- Name – Dies ist ein beschreibender Name für den RCP-File. Der Name sollte nicht länger als 20 Zeichen sein.
- Description (Beschreibung) – Dies ist eine Beschreibung des Files (optional).
- Location where recipe data is stored (applies to all recipe files) (Standort, an dem die Rezeptdaten gespeichert werden (gilt für alle Rezeptfiles)) – Hier können Sie einen Speicherstandort für die RCP-Files bestimmen.
- User Program (Benutzerprogramm) – Sie können Speicherplatz des Benutzerprogramms (Konaktplanlogik) für Rezept-Opera- tionen zuweisen. Sobald Benutzerprogramm-Speicher für den Rezeptgebrauch bestimmt wurde, kann dieser Speicherplatz nicht mehr für Kontaktplanlogik verwendet werden.

**HINWEIS** Der Benutzerprogramm-Speicherplatz kann so geändert werden, dass er nicht mehr für Rezept- operationen, sondern wieder für Kontaktplanlogik genutzt werden kann.

**WICHTIG** Wenn der Benutzerprogramm-Speicher für Rezept- daten verwendet wird, sieht die Verwendung folgendermaßen aus:

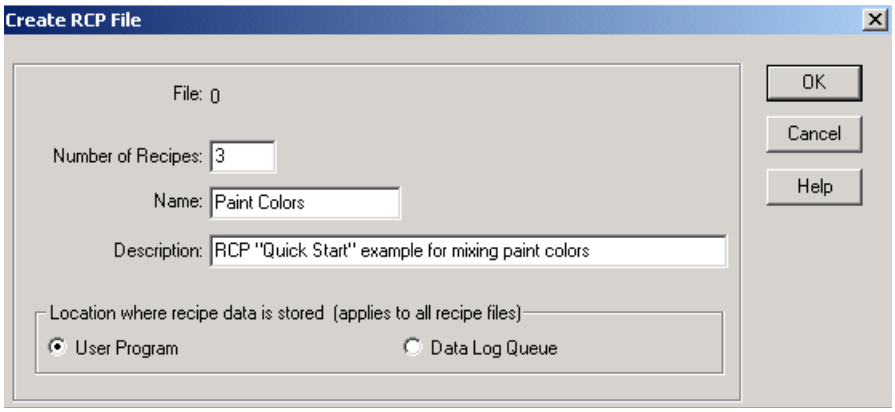
1 K Wörter an Benutzerprogramm-Speicher =
5 K Wörter an Rezeptdatenspeicher

Wie bei der Kontaktplanlogik können die im Benutzerprogramm gespeicherten Rezeptdaten im Speichermodul (1764-MM1, -MM2, -MM1RTC, -MM2RTC) der Steuerung gespeichert werden.

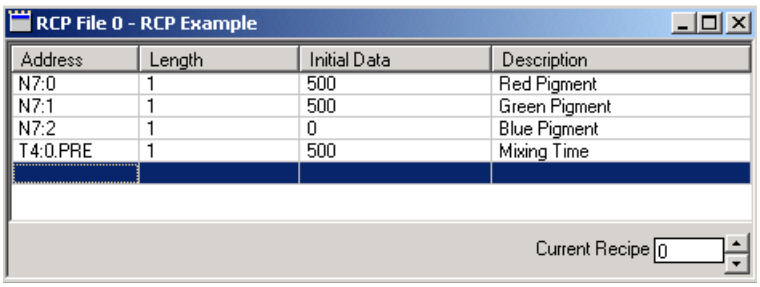
- Data Log Queue (Datenprotokoll-Warteschlange) – Bei 1764-LRP-Prozessoren können Sie Rezeptdaten im Speicherplatz für das Datenprotokoll speichern (48 KB).

**WICHTIG** Im Benutzerprogramm-Speicher gespeicherte Rezeptdaten können im Speichermodul der Steuerung gespeichert werden. Dies ist nicht der Fall für Rezeptdaten,die im Speicher der Datenprotokoll-Warteschlange gespeichert sind. Der Speicher der Datenprotokoll-Warteschlange ist batteriegestützt, kann jedoch nicht auf ein Speichermodul gespeichert werden.

3. Geben Sie wie nachfolgend angegeben die RCP-Fileparameter ein. Klicken Sie am Ende auf *OK*.

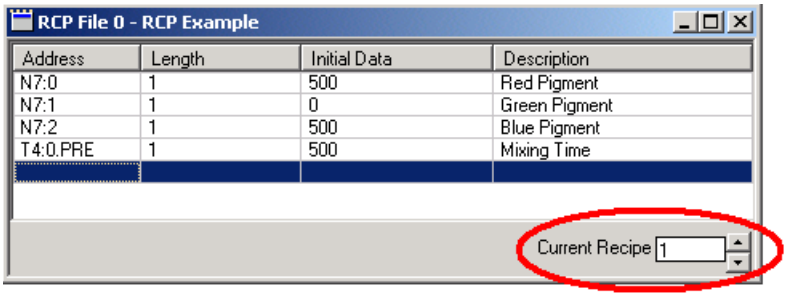



4. Es wird ein neues Fenster angezeigt. Geben Sie in diesem Fenster die nachfolgend genannten Werte ein.

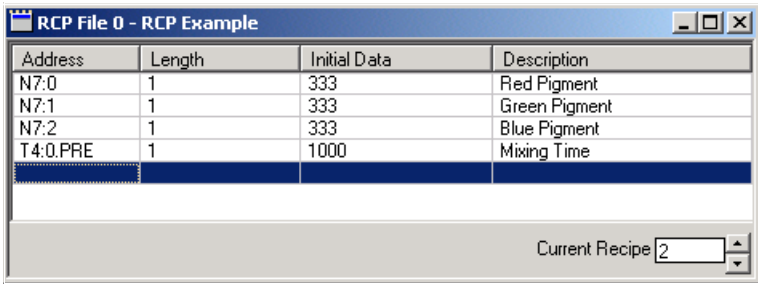



5. Ändern Sie den Wert für das Element *Current Recipe (Derzeitiges Rezept)* von 0 in 1. Beachten Sie, dass zwar die Adresse aber nicht die Daten dupliziert wurde.

**6.** Geben Sie wie nachfolgend angegeben die Daten für Rezept 1 ein.

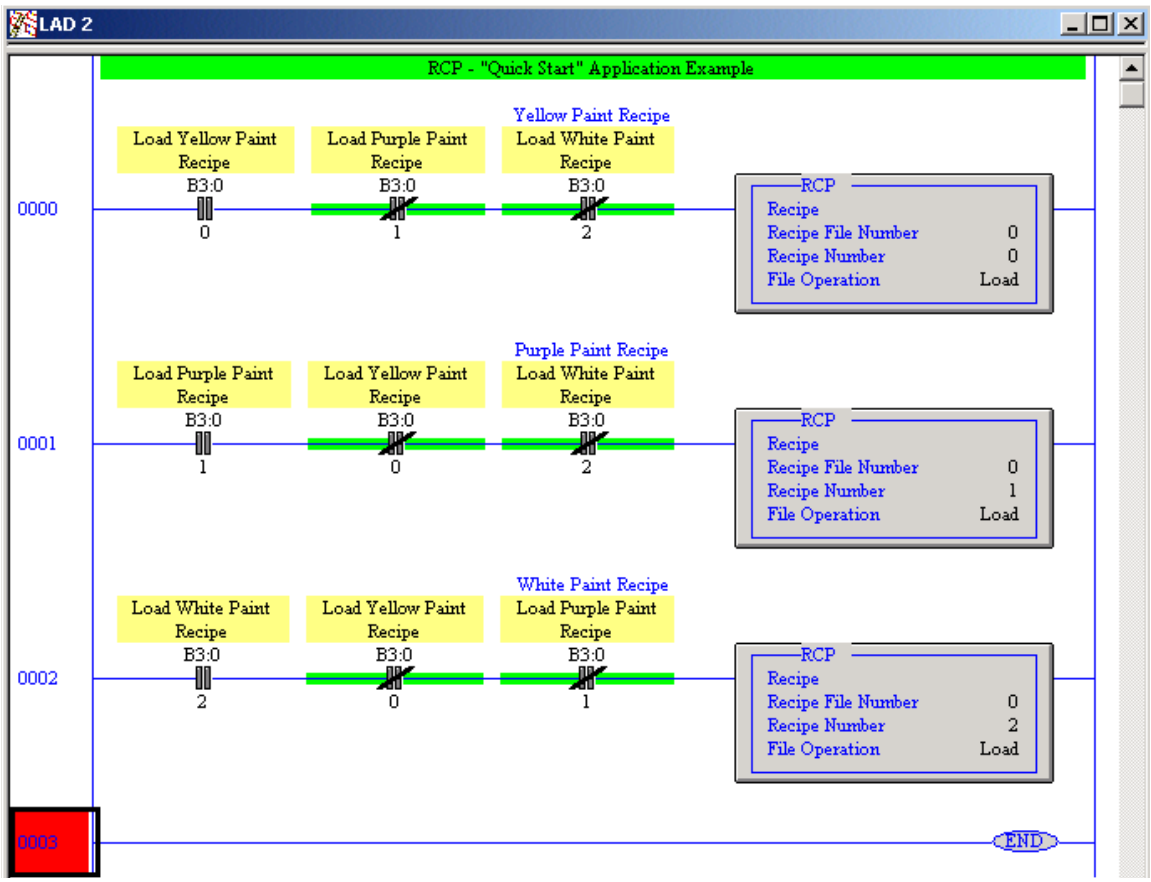



**7.** Wechseln Sie von Rezept 1 zu Rezept 2, und geben Sie die folgenden Daten ein.




Die Rezepte sind nun konfiguriert.

**8.** Erstellen Sie die folgende Kontaktplanlogik.



### *Anwendungserklärung der Operation*

Wenn B3:0/0 aktiviert ist und B3:0/1 sowie B3:0/2 deaktiviert sind, wird Rezeptfile 0:Rezeptnummer 0 ausgeführt und die folgenden Werte werden geladen, sodass die Farbe Gelb erstellt wird.

- N7:0 = 500
- N7:1 = 500
- N7:2 = 0
- T4:0.PRE = 500

Wenn B3:0/1 aktiviert ist und B3:0/0 sowie B3:0/2 deaktiviert sind, wird Rezeptfile 0:Rezeptnummer 1 ausgeführt und die folgenden Werte werden geladen, sodass die Farbe Lila erstellt wird.

- N7:0 = 500
- N7:1 = 0
- N7:2 = 500
- T4:0.PRE = 500

Wenn B3:0/2 aktiviert ist und B3:0/0 sowie B3:0/1 deaktiviert sind, wird Rezeptfile 0:Rezeptnummer 2 ausgeführt, und die folgenden Werte werden geladen, sodass die Farbe Weiß erstellt wird.

- N7:0 = 333
- N7:1 = 333
- N7:2 = 333
- T4:0.PRE = 1000

Überwachen Sie den Datenfile N7. Beachten Sie, dass sich die Werte ändern, nachdem jedes Bit umgeschaltet wurde.

Dieses Beispiel zeigt, wie Werte aus einem RCP-File in eine Datentafeladresse *geladen* werden. Beachten Sie jedoch, dass durch die Änderung der RCP-Fileoperation von einem Ladevorgang in einen Speichernvorgang Werte mithilfe der Kontaktplanlogik in die Rezept- datenbank für jede Rezeptnummer geladen werden können.

## Datenprotokollierung

Die Datenprotokollierung ermöglicht Ihnen das Erfassen (Speichern) von Anwendungsdaten als Datensatz, damit Sie diesen später jederzeit wieder laden können. Jeder Datensatz wird in einer anwenderkonfi- gurierten Warteschlange im batteriegepufferten Speicher (B-RAM) abgelegt. Die Datensätze werden über die Kommunikationsfunktion vom 1764-LRP-Prozessor abgerufen. Dieses Kapitel erläutert, wie die Datenprotokollierung konfiguriert und verwendet wird.

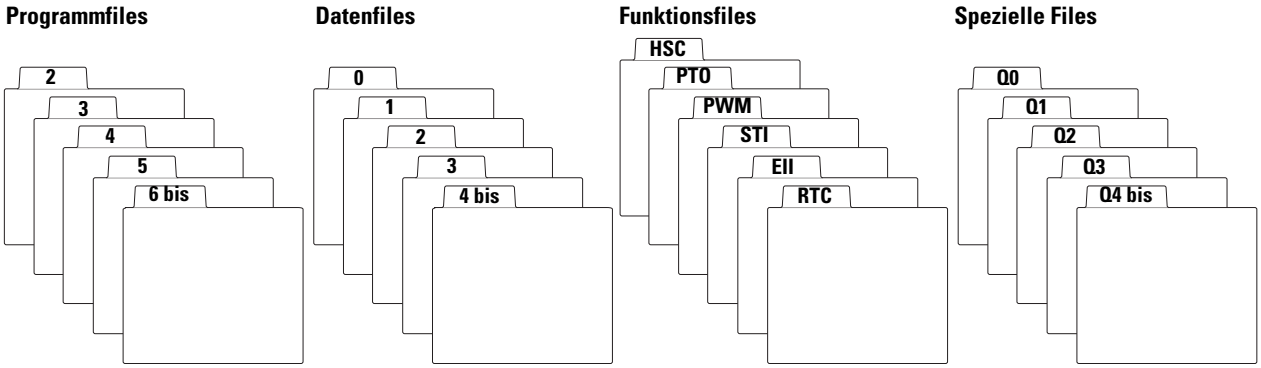
Dieser Abschnitt behandelt folgende Themen:


- „Warteschlangen und Datensätze“ auf Seite 22-7
- „Konfigurieren von Datenprotokoll- Warteschlangen“ auf Seite 22-11
- „DLG – Datenprotokollbefehl“ auf Seite 22-13
- „Datenprotokoll-Statusfile“ auf Seite 22-14
- „Laden (Lesen) von Datensätzen“ auf Seite 22-16


## Warteschlangen und Datensätze

Der 1764-LRP-Prozessor bietet für die Datenprotokollierung 48 KB (48 x 1024) zusätzlichen Speicher. Innerhalb dieses Speichers können Sie bis zu 256 (0 bis 255) Datenprotokollierungs-Warteschlangen definie- ren. Jede Warteschlange lässt sich nach Größe (maximale Anzahl der gespeicherten Datensätze) und Länge (jeder Datensatz hat zwischen 1 und 80 Zeichen) konfigurieren. Länge und maximale Anzahl von Datensätzen bestimmen, wie viel Speicher von der Warteschlange belegt werden. Sie können eine große Warteschlange oder mehrere kleine Warteschlangen definieren.

Der für die Datenprotokollierung verwendete Speicher ist unabhängig vom übrigen Prozessorspeicher. Auf ihn kann nicht über das Anwenderprogramm zugegriffen werden. Jeder Datensatz wird beim Ausführen des Befehls gespeichert und ist nicht flüchtig (batteriege- puffert). Auf diese Weise gehen beim Ausschalten keine Daten verloren.



## Beispielwarteschlange 0

Diese Warteschlange soll veranschaulichen, wie die Zeichenketten- länge jedes Datensatzes und die maximale Anzahl der Datensätze berechnet wird.

**Tabelle 22.2 Warteschlange 0 (Datum = ✔, Zeit = ✔, Endezeichen = ,)**

| | Datum | | Zeit | | N7:11 | | L14:0 | | T4:5.ACC | | I1:3.0 | | B3:2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Datensatz 0 | 01/10/2000 | , | 20:00:00 | , | 2315 | , | 103457 | , | 200 | , | 8190 | , | 4465 |
| Datensatz 1 | 01/10/2000 | , | 20:30:00 | , | 2400 | , | 103456 | , | 250 | , | 8210 | , | 4375 |
| Datensatz 2 | 01/10/2000 | , | 21:00:00 | , | 2275 | , | 103455 | , | 225 | , | 8150 | , | 4335 |
| Datensatz 3 | 01/10/2000 | , | 21:30:00 | , | 2380 | , | 103455 | , | 223 | , | 8195 | , | 4360 |
| Datensatz 4 | 01/10/2000 | , | 22:00:00 | , | 2293 | , | 103456 | , | 218 | , | 8390 | , | 4375 |
| Datensatz 5 | 01/10/2000 | , | 22:30:00 | , | 2301 | , | 103455 | , | 231 | , | 8400 | , | 4405 |
| Datensatz 6 | 01/10/2000 | , | 23:00:00 | , | 2308 | , | 103456 | , | 215 | , | 8100 | , | 4395 |
| Datensatz 7 | 01/10/2000 | , | 23:30:00 | , | 2350 | , | 103457 | , | 208 | , | 8120 | , | 4415 |
| Datensatz 8 | 01/11/2000 | , | 00:00:00 | , | 2295 | , | 103457 | , | 209 | , | 8145 | , | 4505 |
| Datensatz 9 | 01/11/2000 | , | 00:30:00 | , | 2395 | , | 103456 | , | 211 | , | 8190 | , | 4305 |
| Datensatz 10 | 01/11/2000 | , | 01:00:00 | , | 2310 | , | 103455 | , | 224 | , | 8195 | , | 4455 |
| Datensatz 11 | 01/11/2000 | , | 01:30:00 | , | 2295 | , | 103456 | , | 233 | , | 8190 | , | 4495 |

### *Zeichenkettenlänge des Datensatzes*

Die Größe eines Datensatzes ist begrenzt, sodass die Länge der maximalen formatierten Zeichenkette 80 Zeichen nicht überschreitet. Anhand der folgenden Tabelle lässt sich die Länge der formatierten Zeichenkette bestimmen.

| Daten | Belegter Speicher | Größe der formatierten Zeichenkette |
|---|---|---|
| Endezeichen | 0 Byte | Zeichen 1 |
| Wort | 2 Byte | 6 Zeichen |
| Doppelwort | 4 Byte | 11 Zeichen |
| Datum | 2 Byte | 10 Zeichen |
| Zeit | 2 Byte | 8 Zeichen |

Für Warteschlange 0 liegt die Länge der formatierten Zeichenkette bei 59 Zeichen (siehe unten):

| Daten | Datum | | Zeit | | N7:11 | | L14:0 | | T4:5.ACC | | I1:3.0 | | I1:2.1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Zeichen | 10 | 1 | 8 | 1 | 6 | 1 | 11 | 1 | 6 | 1 | 6 | 1 | 6 |

= 10 + 1 + 8 + 1 + 6 + 1 + 11 + 1 + 6 + 1 + 6 + 1 + 6
= 59 Zeichen

*Number of Records (Anzahl der Datensätze)*

In Warteschlange 0 belegt jeder Datensatz in diesem Beispiel:

| Datensatzfeld | Speicherbelegung |
|---|---|
| Datum | 2 Byte |
| Zeit | 2 Byte |
| N7:11 | 2 Byte |
| L14:0 | 4 Byte |
| T4:5.ACC | 2 Byte |
| I1:3.0 | 2 Byte |
| B3:2 | 2 Byte |
| Integritätsüberprüfung | 2 Byte |
| **Summe** | **18 Byte** |

In diesem Beispiel belegt jeder Datensatz 18 Byte. Wenn also eine Warteschlange konfiguriert wurde, könnten maximal 2730 Datensätze gespeichert werden. Die maximale Anzahl von Datensätzen wird wie folgt berechnet:

Maximale Anzahl von Datensätzen = Größe des Datenprotokollfiles/Datensatzgröße
= 48 KB/18 Byte
= (48)(1024)/18
= 2730 Datensätze

## Beispielwarteschlange 5

**Tabelle 22.3 Warteschlange 5 (Zeit = ✔, Endezeichen = TAB)**

| | Zeit | | N7:11 | | I1:3.0 | | I1:2.1 |
|---|---|---|---|---|---|---|---|
| Datensatz 0 | 20:00:00 | TAB | 2315 | TAB | 8190 | TAB | 4465 |
| Datensatz 1 | 20:30:00 | TAB | 2400 | TAB | 8210 | TAB | 4375 |
| Datensatz 2 | 21:00:00 | TAB | 2275 | TAB | 8150 | TAB | 4335 |
| Datensatz 3 | 21:30:00 | TAB | 2380 | TAB | 8195 | TAB | 4360 |
| Datensatz 4 | 22:00:00 | TAB | 2293 | TAB | 8390 | TAB | 4375 |
| Datensatz 5 | 22:30:00 | TAB | 2301 | TAB | 8400 | TAB | 4405 |
| Datensatz 6 | 23:00:00 | TAB | 2308 | TAB | 8100 | TAB | 4395 |

*Zeichenkettenlänge des Datensatzes*

Die Größe eines Datensatzes ist begrenzt, sodass die Länge der maximalen formatierten Zeichenkette 80 Zeichen nicht überschreitet. Anhand der folgenden Tabelle lässt sich die Länge der formatierten Zeichenkette bestimmen.

| Daten | Belegter Speicher | Größe der formatierten Zeichenkette |
|---|---|---|
| Endezeichen | 0 Byte | Zeichen 1 |
| Wort | 2 Byte | 6 Zeichen |
| Doppelwort | 4 Byte | 11 Zeichen |
| Datum | 2 Byte | 10 Zeichen |
| Zeit | 2 Byte | 8 Zeichen |

Für Warteschlange 5 liegt die Länge der formatierten Zeichenkette bei 29 Zeichen (siehe unten):

| Daten | Zeit | | N7:11 | | I1:3.0 | | I1:2.1 |
|---|---|---|---|---|---|---|---|
| Zeichen | 8 | 1 | 6 | 1 | 6 | 1 | 6 |

= 8 + 1 + 6 + 1 + 6 + 1 + 6 = 29 Zeichen

### *Number of Records (Anzahl der Datensätze)*

In Warteschlange 5 belegt jeder Datensatz in diesem Beispiel:

| **Datensatzfeld** | **Speicherbelegung** |
|---|---|
| Zeit | 2 Byte |
| N7:11 | 2 Byte |
| I1:3.0 | 2 Byte |
| I1:2.1 | 2 Byte |
| Integritätsüberprüfung | 2 Byte |
| **Summe** | **10 Byte** |

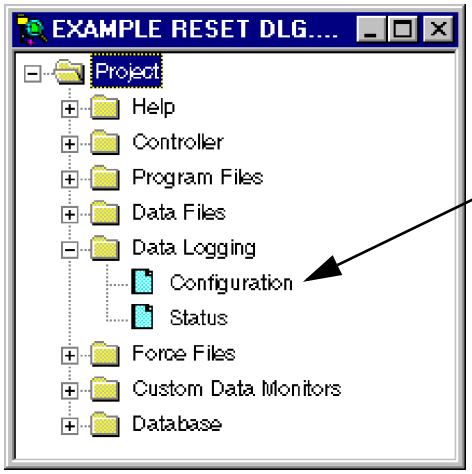
Jeder Datensatz belegt 10 Byte. Wenn also nur eine Warteschlange konfiguriert wurde, könnten maximal 4915 Datensätze gespeichert werden. Die maximale Anzahl von Datensätzen wird wie folgt berechnet:

Maximale Anzahl von Datensätzen = Größe des Datenprotokollfiles/Datensatzgröße
= 48 KB/10 Byte
= (48)(1024)/10
= 4915 Datensätze

## Konfigurieren von Datenprotokoll-Warteschlangen

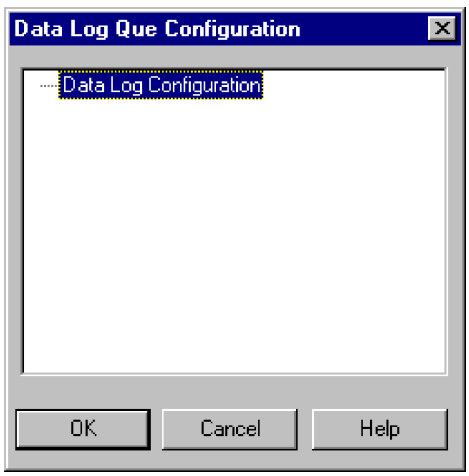
Die Datenprotokollierung wird mit Hilfe der RSLogix 500-Programmiersoftware, Version V4.00.00 oder höher, konfiguriert.

1. Öffnen Sie eine 1764-LRP-Anwendung. Der erste Schritt bei der Datenprotokollierung ist die Konfigurierung der Datenprotokoll-Warteschlange(n). Zugriff auf diese Funktion erhalten Sie über den RSLogix 500-Projektbaum:



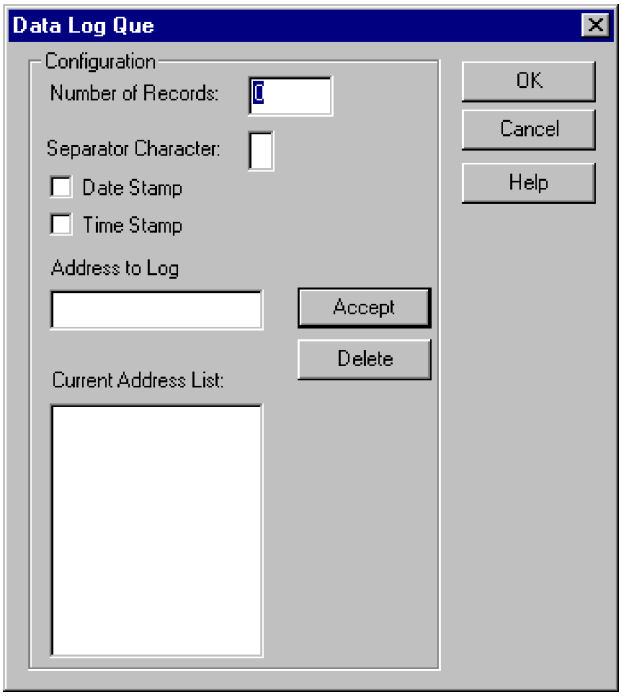

Doppelklicken Sie auf die Option *Configuration* (Konfiguration), um auf die Datenprotokoll-Konfiguration zuzugreifen.

2. Das Fenster „Data Log Que“ (Datenprotokollwarteschlange) wird angezeigt. Klicken Sie doppelt auf die Option „Data Log Configuration“ (Datenprotokollkonfiguration).




Das Konfigurationsfenster „Data Log Que“ (Datenprotokollwarteschlange) vor dem Erstellen einer Warteschlange.

3. Das Dialogfenster „Data Log Que“ (Datenprotokollwarteschlange) wird aufgerufen (siehe unten). Geben Sie über dieses Dialog- fenster die Warteschlangendaten ein.




Geben Sie Folgendes ein:

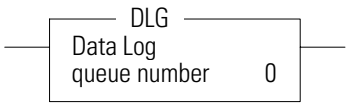
| Konfigurationsparameter für die Datenprotokoll-Warteschlange | Beschreibung |
|---|---|
| Number of Records (Anzahl der Datensätze) | Definiert die Anzahl der Datensätze in der Warteschlange. |
| Separator Character (Trennzeichen) | Wählen Sie das Zeichen aus, das in dieser Warteschlange als Trennzeichen dienen soll (Tabulator, Komma oder Leerzeichen). Das Trennzeichen kann in allen Warteschlangen identisch oder unterschiedlich sein. |
| Date Stamp (Datumsstempel) – optional | Falls ausgewählt, wird das Datum im Format mm/tt/jjjj protokolliert(1). |
| Time Stamp (Zeitstempel) – optional | Falls ausgewählt, wird die Uhrzeit im Format hh:mm:ss protokolliert(1). |
| Address to Log (Zu protokollierende Adresse) | Geben Sie die Adresse eines aufzuzeichnenden Elements ein, und klicken Sie auf die Option zum Annehmen, um die Adresse der aktuellen Adressliste hinzuzufügen. Die Adresse kann aus 16- oder 32-Bit-Daten bestehen. |
| Current Address List (Aktuelle Adressliste) | Diese Liste enthält die aufzuzeichnenden Elemente. Die Datensatzgröße kann maximal 80 Byte betragen. Klicken Sie zum Löschen von Elementen aus dieser Liste auf die entsprechende Schaltfläche. Weitere Informationen zur Datensatzgröße finden Sie auf Seite 22-8. |

*Ein Datensatz besteht aus konfiguriertem Datumsstempel, Zeitstempel, aktueller Adressliste und Trennzeichen.*

(1) Verfügt die Steuerung nicht über eine Echtzeituhr, und sind die Optionen „Date Stamp" (Dateumsmarke) und „Time Stamp" (Zeitmarke) ausgewählt (aktiviert), wird das Datum mit 00/00/0000 und die Uhrzeit mit 00:00:00 protokolliert.

4. Wenn Sie alle Daten für die Datenprotokoll-Warteschlange einge- geben haben, klicken Sie auf OK. Die Warteschlange wird im Fenster „Data Log Que" (Datenprotokollwarteschlange) mit einer entsprechenden Warteschlangennummer eingefügt. Diese Warteschlangennummer muss im DLG-Befehl verwendet werden.

# DLG – Datenprotokollbefehl




Befehlstyp: Ausgang

**Tabelle 22.4 Ausführungszeit des DLG-Befehls**

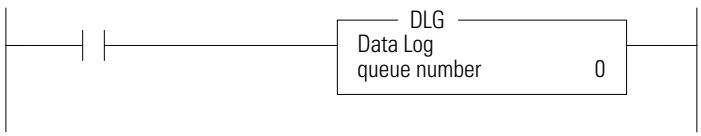
| Steuerung | Strompfad | |
|---|---|---|
| | wahr | unwahr |
| MicroLogix 1500 1764-LRP | 67,5 µs + 11,8 µs/ Datumsstempel<br>+ 12,4 µs/Zeitstempel<br>+ 9,1 µs/protokolliertes Wort<br>+ 16,2 µs/protokolliertes Doppelwort | 6,7 µs |

**WICHTIG** Sie müssen eine Datenprotokoll-Warteschlange vor der Programmierung eines DLG-Befehls in Ihrem Kontaktplanprogramm konfigurieren.

Der DLG-Befehl triggert die Speicherung eines Datensatzes. Der DLG-Befehl verfügt über einen Operanden:

**Queue Number (Warteschlangennummer)** – Gibt an, welche Datenprotokollwarteschlange einen Datensatz erfasst.

Nur der DLG-Befehl erfasst Daten beim Übergang des Strompfads von unwahr nach wahr. Der DLG-Strompfad muss zurückgesetzt (unwahr abgefragt) werden, bevor er erneut Daten erfassen kann. Der DLG-Befehl darf nie alleine in einem Strompfad platziert werden. Ihm muss stets eine Logik vorausgehen (siehe unten):



## Datenprotokoll-Statusfile

Für jede Datenprotokoll-Warteschlange gibt es einen Datenprotokoll-Statusfile (DLS). Der DLS-File steht erst nach der Konfiguration einer Datenprotokoll-Warteschlange zur Verfügung.

Der Datenprotokoll-Statusfile verfügt über 3-Wort-Elemente. Wort 0 ist über die Kontaktplanlogik nur nach Bit adressierbar. Die Worte 1 und 2 sind über die Kontaktplanlogik nach Wort und/oder Bit adressierbar.

Die Anzahl der DLS-Fileelemente hängt von der Anzahl der in der Anwendung angegebenen Warteschlangen ab. Status-Bits und Worte sind nachfolgend beschrieben.

**Tabelle 22.5 Elemente des Datenprotokoll-Statusfiles (DLS)**

| | **Steuerelement** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Wort** | **15** | **14** | **13** | **12** | **11** | **10** | **09** | **08** | **07** | **06** | **05** | **04** | **03** | **02** | **01** | **00** |
| 0 | EN(1) | 0 | DN (2) | OV(3) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | FSZ = Filegröße (Anzahl der zugewiesenen Datensätze) | | | | | | | | | | | | | | | |
| 2 | RST = Gespeicherte Datensätze (Anzahl der protokollierten Datensätze) | | | | | | | | | | | | | | | |

(1) EN = Aktiv-Bit

(2) DN = Fertig-Bit

(3) OV = Überlauf-Bit

### *Datenprotokollierung aktivieren (EN)*

Ist der DLG-Befehlsstrompfad wahr, wird das EN-Bit zum Aktivieren der Datenprotokollierung gesetzt (1), und der DLG-Befehl protokol- liert den definierten Datensatz. Zum Adressieren dieses Bits in der Kontaktplanlogik ist folgendes Format zu verwenden: DLS0:Q/EN, wobei Q die Warteschlangennummer darstellt.

### *Datenprotokollierung fertig (EN)*

Das Fertig-Bit wird verwendet, um anzuzeigen, wann die zugewiesene Warteschlange voll ist. Dieses Bit wird durch den DLG-Befehl gesetzt (1), wenn die Warteschlange voll ist. Dieses Bit wird gelöscht, wenn ein Datensatz aus der Warteschlange gelöscht wird. Zum Adressieren dieses Bits in der Kontaktplanlogik ist folgendes Format zu verwen- den: DLS0:Q/DN, wobei Q die Warteschlangennummer darstellt.

### *Datenprotokollierungs-Überlauf (OV)*

Das Datenprotokollierungs-Überlauf-Bit (OV) wird verwendet, um anzuzeigen, wann ein Datensatz in der zugewiesenen Warteschlange überschrieben wird. Dieses Bit wird durch den DLG-Befehl gesetzt (1), wenn ein Datensatz überschrieben wird. Das OV-Bit bleibt so lange gesetzt, bis Sie es löschen (0). Zum Adressieren dieses Bits in der Kontaktplanlogik ist folgendes Format zu verwenden: DLS0:Q/OV, wobei Q die Warteschlangennummer darstellt.

### *Filegröße (FSZ)*

Das FSZ-Bit zeigt die Anzahl der Datensätze an, die dieser Warte- schlange zugewiesen sind. Die Anzahl der Datensätze wird nach dem Konfigurieren der Warteschlange festgelegt. FSZ kann mit RST verwendet werden, um festzustellen, wie voll die Warteschlange ist. Zum Adressieren dieses Worts in der Kontaktplanlogik ist folgendes Format zu verwenden: DLS0:Q.FSZ, wobei Q die Warteschlangennummer darstellt.

### *Gespeicherte Datensätze (RST)*

Das RST-Bit gibt an, wie viele Datensätze sich in der Warteschlange befinden. RST wird vermindert, wenn ein Datensatz aus einem Kommunikationsgerät gelesen wird. Zum Adressieren dieses Worts in der Kontaktplanlogik ist folgendes Format zu verwenden: DLS0:Q/RST, wobei Q die Warteschlangennummer darstellt.

**HINWEIS** Soll ein weiterer Datensatz in einer vollen Warte- schlange gespeichert werden, wird der älteste Datensatz überschrieben. Das Verhalten der Warte- schlange ist identisch mit dem des FIFO-Stapels – First in, First out. Soll ein weiterer Datensatz in einer vollen Warteschlange gespeichert werden, wird der „erste“ Datensatz gelöscht.

Die DLS-Daten können in folgenden Befehlstypen verwendet werden:

| **Befehlstyp:** | **Operand** |
|---|---|
| Relais (Bit) | Zielausgangs-Bit |
| Compare | Source (Quelle) A |
| | Source (Quelle) B |
| | Unterer Grenzwert (LIM-Befehl) |
| | Test (LIM-Befehl) |
| | Oberer Grenzwert (LIM-Befehl) |
| | Quelle (MEQ-Befehl) |
| | Maske (MEQ-Befehl) |
| | Vergleich (MEQ-Befehl) |
| Mathematische Befehle | Source (Quelle) A |
| | Source (Quelle) B |
| | Eingang (SCP-Befehl) |
| Logik | Source (Quelle) A |
| | Source (Quelle) B |
| Übertragung | Quelle |

## Laden (Lesen) von Datensätzen

Daten werden aus einer Datenprotokollierungs-Warteschlange geladen, indem ein logischer Lesebefehl gesendet wird, der den Ladefile des Datenprotokolls adressiert. Der älteste Datensatz wird zuerst geladen und anschließend gelöscht. Der Datensatz wird gelöscht, sobald er zur Übertragung in die Warteschlange gestellt wurde. Sollte vor Abschluss der Übertragung eine Unterbrechung der Stromversorgung auftreten, geht der Datensatz verloren.

Die Daten werden als ASCII-Zeichensatz mit folgendem Format geladen:

**<Datum><UDS><Zeit><UDS><1. Datensatz><UDS><2. Datensatz><UDS>…<UDS><Letzter Datensatz><NUL>**

- Wobei:
  <date> = mm/tt/jjjj - ASCII-Zeichen (Datum ist optional)
  <time> = hh:mm:ss - ASCII-Zeichen (Zeit ist optional)
  <UDS> = Anwenderdefiniertes Trennzeichen (TAB, KOMMA oder LEERZEICHEN)
  <X Data> = Dezimale ASCII-Darstellung des Werts der Daten
  <NUL> = Datensatzzeichenkette ist mit einer Null abgeschlossen
- Verfügt die Steuerung nicht über ein Echtzeituhrmodul, wird <Datum> als 00/00/0000 und <Zeit> als 00:00:00 formatiert.
- Das Kommunikationsgerät bestimmt die Anzahl der Datensätze, die aufgezeichnet, doch nicht geladen wurden. Siehe den Abschnitt „Datenprotokoll-Statusfile“ auf Seite 22-14.
- Die Steuerung führt für jeden Datensatz eine Integritätsüberprüfung aus. Ist die Datenintegritäts-überprüfung ungültig, wird an das Kommunikationsgerät eine Fehlerantwort gesendet. Der Datensatz wird gelöscht, sobald die Fehlerantwort zur Übertragung in die Warteschlange gestellt wurde.

**HINWEIS** Zur einfachen Verwendung mit Microsoft Excel verwenden Sie als Trennzeichen das TAB-Zeichen (Tabulator).

## Zugreifen auf den Ladefile

Verwenden Sie ein spezielles Lade-Tool oder Ihre eigene Anwendung.

### Lade-Tools

Es gibt zahlreiche Lade-Tools, die zur Verwendung mit Palm™ OS, Windows™ CE, Windows 9x und Windows NT geeignet sind. Diese kostenlosen Tools können Sie von unserer Website unter folgender Adresse herunterladen: http://www.ab.com/micrologix.

## Informationen zum Erstellen Ihrer eigenen Anwendung

*Steuerung empfängt Kommunikationspaket*

**Tabelle 22.6 Befehlsstruktur**

| DST | SRC | CMD 0f | STS | TNS | FNC A2 | Bytegröße | Filenr. | Filetyp | Ele.-Nr. | S/Ele.-Nr. |
|---|---|---|---|---|---|---|---|---|---|---|

| Feld | Funktion | Beschreibung |
|---|---|---|
| DST | Zielknoten | |
| SRC | Quellenknoten | |
| CMD | Befehlscode | |
| STS | Statuscode | Auf Null (0) gesetzt |
| TNS | Übertragungsnummer | Immer 2 Byte |
| FNC | Funktionscode | |
| Bytegröße | Anzahl der zu lesenden Bytes | Länge der formatierten Zeichenkette (siehe nachstehende Gleichung) |
| Filenummer | | Immer auf Null (0) gesetzt |
| Filetyp | | Muss A5 (hex) sein |
| Elementnummer | Warteschlangennummer | Bestimmt die zu lesende Warteschlange (0 bis 255) |
| Unter-/Elementnummer | | Immer auf Null (0) gesetzt |

**Tabelle 22.7 Gleichung**

| | | | | | | |
|---|---|---|---|---|---|---|
| Datensatzfeld 1 | + Datensatzfeld 2 | + Datensatzfeld 3 | . . . | + Datensatzfeld 7 | = | Länge der formatierten Zeichenkette |

**Tabelle 22.8 Datensatzfeldgrößen**

| Datentyp | Maximale Größe |
|---|---|
| Wort | 7 Byte (Zeichen) |
| Doppelwort | 12 Byte (Zeichen) |
| Datumsfeld | 11 Byte (Zeichen) |
| Zeitfeld | 9 Byte (Zeichen) |

**HINWEIS** Die Länge der formatierten Zeichenkette darf maximal 80 Byte betragen.

**HINWEIS** Das letzte Byte ist ein Nullwert, der das Endezeichen darstellt.

*Steuerung antwortet*

**Tabelle 22.9 Antwortstruktur**

| SRC | DST | CMD 4f | STS | TNS | DATA | EXT STS |
|---|---|---|---|---|---|---|

| Feld | Funktion | Beschreibung |
|---|---|---|
| SRC | Quellenknoten | |
| DST | Zielknoten | |
| CMD | Befehlscode | |
| STS | Statuscode | |
| TNS | Übertragungsnummer | Immer 2 Byte |
| DATA | | Formatierte Zeichenkette |

Bei nicht erfolgreicher Integritätsüberprüfung wird der Datensatz gelöscht und ein Fehler mit STS von 0xF0 und ext STS von 0x0E gesendet.

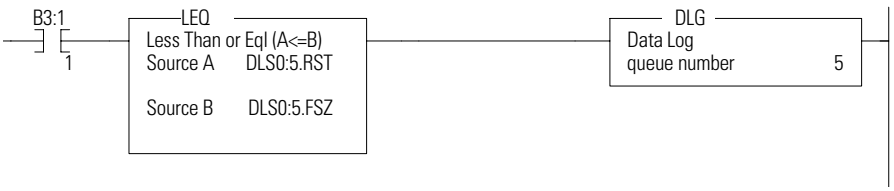
Weitere Informationen zum Schreiben eines DF1-Protokolls finden Sie in der Allen-Bradley-Publikation 1770-6.5.16, *DF1 Protocol and Command Set Reference Manual* (erhältlich im Internet über www.theautomationbookstore.com).

## Bedingungen, die mit dem Datenladefile auftreten

**WICHTIG** Die Daten im Ladefile können nur einmal gelesen werden. Anschließend werden sie durch den Prozessor gelöscht.

Die folgenden Bedingungen führen zum Verlust der zuvor protokollierten Daten:

- Herunterladen eines Programms von RSLogix 500 zur Steuerung.
- Übertragung des Speichermoduls an die Steuerung (*Ausnahme ist das automatische Laden desselben Programms durch das Speichermodul*).
- Volle Warteschlange - Bei einer vollen Warteschlange überschrei- ben die neuen Datensätze die vorhandenen Datensätze (begin- nend am Anfang des Files). Um dies zu verhindern, fügen Sie fol- genden Strompfad in Ihr Kontaktplanprogramm ein:



# MicroLogix 1200 – Speicherbelegung und Befehlsausführungszeit

Dieser Anhang enthält eine vollständige Liste der MicroLogix 1200-Programmierbefehle. In dieser Liste sind die Speicherbelegung und die Ausführungszeit für jeden Befehl aufgeführt. Außerdem wird die Ausführungszeit bei indirekter Adressierung angegeben und ein Arbeitsblatt zur Berechnung der Abfragezeit bereitgestellt.

## Speicherbelegung und Ausführungszeit von Programmierbefehlen

Die folgende Tabelle zeigt Ausführungsdauer und Speicherbedarf der einzelnen Steuerungsbefehle. Diese Werte sind abhängig davon, ob als Datenformat *Worte* oder *Doppelworte* verwendet werden.

.

**Tabelle A.1 Speicherbelegung und Ausführungszeiten der MicroLogix 1200-Programmierbefehle**

| Programmierbefehle | Befehls-mnemonik | Wort | | | Doppelwort | | |
|---|---|---|---|---|---|---|---|
| | | Ausführungszeit in µs | | Speicher-belegung in Worten | Ausführungszeit in µs | | Speicher-belegung in Worten |
| | | unwahr | wahr | | unwahr | wahr | |
| **4 in 1 aus 16 dekodieren** | **DCD** | 0,0 | 1,9 | 1,9 | Keine Doppelwortadressierung. | | |
| **Absolutwert** | **ABS** | 0,0 | 3,8 | | | | |
| **Addition** | **ADD** | 0,0 | 2,7 | 3,3 | 0,0 | 11,9 | 3,5 |
| **Anwender-Interrupt aktivieren** | **UIE** | 0,0 | 0,8 | 0,9 | Keine Doppelwortadressierung. | | |
| **Anwender-Interrupt deaktivieren** | **UID** | 0,0 | 0,8 | 0,9 | | | |
| **Anwender-Interrupt entfernen** | **UIF** | 0,0 | 12,3 | 0,9 | | | |
| **Anzahl der ASCII-Zeichen im Puffer(1)** | **ACB** | 12,1 | 103,1 | 3,3 | | | |
| **ASCII-Ganzzahl in Zeichenkette** | **AIC** | 0,0 | 29,3 +5,2/ Zeichen | 1,4 | 0,0 | 82,0 | 1,6 |
| **ASCII schreiben** | **AWT** | 14,1 | 268 + 12/ Zeichen | 3,4 | Keine Doppelwortadressierung. | | |
| **ASCII schreiben und anhängen** | **AWA** | 14,1 | 268 + 12/ Zeichen | 3,4 | | | |
| **ASCII-Handshake-Leitungen(1)** | **AHL** | 11,9 | 109,4 | 5,3 | | | |
| **ASCII-Lesen(1)** | **ARD** | 11,8 | 132,3 + 49,7/ Zeichen | 4,3 | | | |
| **ASCII-Puffer löschen** | **ACL** | 0,0 | löschen: beide 249,1 empfangen 28,9 übertragen 33,6 | 1,2 | | | |
| **ASCII-Testpuffer für Zeile(1)** | **ABL** | 12,5 | 115 + 8,6/ Zeichen | 3,3 | | | |
| **ASCII-Zeichenkette extrahieren(1)** | **AEX** | 0,0 | 14,8 + 2,9/ Zeichen | 2,5 | | | |

**Tabelle A.1 Speicherbelegung und Ausführungszeiten der MicroLogix 1200-Programmierbefehle**

| Programmierbefehle | Befehls-mnemonik | Wort | | | Doppelwort | | |
|---|---|---|---|---|---|---|---|
| | | Ausführungszeit in µs | | Speicher-belegung in Worten | Ausführungszeit in µs | | Speicher-belegung in Worten |
| | | unwahr | wahr | | unwahr | wahr | |
| **ASCII-Zeichenkette in Ganzzahl(1)** | **ACI** | 0,0 | 17,6 + 7,2/ Zeichen | 1,5 | 0,0 | 24,6 + 11,6/Zeichen | 1,5 |
| **ASCII-Zeichenkette suchen(1)** | **ASC** | 0,0 | 16,2 + 4,0/ übereinst. Zeichen | 6,0 | Keine Doppelwortadressierung. | | |
| **ASCII-Zeichenkette vergleichen(1)** | **ASR** | 0,0 | 9,2 + 4,0/ übereinst. Zeichen | 1,8 | | | |
| **ASCII-Zeichenkette verketten(1)** | **ACN** | 0,0 | 22,6 + 11,5/ Zeichen | 2,0 | | | |
| **ASCII-Zeile lesen(1)** | **ARL** | 11,7 | 139,7 + 50,1/ Zeichen | 4,3 | Keine Doppelwortadressierung. | | |
| **Auf geschlossen prüfen** | **XIC** | 0,8 | 0,9 | 1,0 | | | |
| **Auf offen prüfen** | **XIO** | 0,8 | 0,9 | 1,0 | | | |
| **Ausgang einschalten** | **OTE** | 1,1 | 1,4 | 1,6 | | | |
| **Ausgang rücksetzen** | **OTU** | 0,0 | 1,1 | 0,6 | | | |
| **Ausgang setzen** | **OTL** | 0,0 | 1,0 | 0,6 | | | |
| **Bit nach links verschieben** | **BSL** | 1,3 | 32 + 1,3/Wort | 3,8 | | | |
| **Bit nach rechts verschieben** | **BSR** | 1,3 | 32 + 1,3/Wort | 3,8 | | | |
| **Byte-Tausch(1)** | **SWP** | 0,0 | 13,7 + 2,2/ vertauschtem Wort | 1,5 | | | |
| **Division** | **DIV** | 0,0 | 12,2 | 2,0 | 0,0 | 42,8 | 3,5 |
| **E/AAuffrischung** | **REF** | 0,0 | siehe S. A-7 | 0,5 | Keine Doppelwortadressierung. | | |
| **Echtzeituhr anpassen** | **RTA** | 3,7 | 4,7 (556,2 wahr-zu-unwahr-Wechsel) | | | | |
| **Einzelimpuls** | **ONS** | 1,9 | 2,6 | 3,5 | | | |
| **ExklusivOder** | **XOR** | 0,0 | 3,0 | 2,8 | 0,0 | 9,9 | 3,0 |
| **FIFO entladen** | **FFU** | 10,4 | 33 + 0,8/Wort | 3,4 | 10,4 | 36 + 1,5/Doppelwort | 3,4 |
| **FIFO laden** | **FFL** | 11,1 | 11,3 | 3,4 | 11,2 | 11,7 | 3,9 |
| **File kopieren** | **COP** | 0,0 | 19 + 0,8/Wort | 2,0 | Keine Doppelwortadressierung. | | |
| **Filefüllung** | **FLL** | 0,0 | 14 + 0,6/Wort | 2,0 | 0,0 | 15 + 1,2/Doppelwort | 2,5 |
| **Gleich** | **EQU** | 1,1 | 1,3 | 1,3 | 1,9 | 2,8 | 2,6 |
| **Gray-Code** | **GCD** | 0,0 | 9,5 | | Keine Doppelwortadressierung. | | |
| **Grenzwerttest** | **LIM** | 6,1 | 6,4 | 2,3 | 13,6 | 14,4 | 4,0 |
| **Größer als** | **GRT** | 1,1 | 1,3 | 1,3 | 2,7 | 2,8 | 2,4 |
| **Größer als oder gleich** | **GEQ** | 1,1 | 1,3 | 1,3 | 2,7 | 2,8 | 2,9 |
| **Hauptsteuerbefehl** | **MCR (Start)** | 1,2 | 1,2 | 1,0 | Keine Doppelwortadressierung. | | |
| | **MCR (Ende)** | 1,6 | 1,6 | 1,5 | | | |
| **Hochgeschwindigkeitsladen** | **HSL** | 0,0 | 46,7 | 7,3 | 0,0 | 47,3 | 7,8 |
| **Impulsausgang(1)** | **PTO** | 24,4 | 85,6 | 1,9 | Keine Doppelwortadressierung. | | |
| **In BCD** | **TOD** | 0,0 | 17,2 | 1,8 | | | |
| **Interrupt Subroutine** | **INT** | 1,0 | 1,0 | 0,3 | | | |
| **Istwert rücksetzen** | **RAC** | Keine Wortadressierung. | | | 0,0 | 21,2 | 2,0 |
| **Kleiner als** | **LES** | 1,1 | 1,3 | 1,3 | 2,7 | 2,8 | 2,9 |

**Tabelle A.1 Speicherbelegung und Ausführungszeiten der MicroLogix 1200-Programmierbefehle**

| Programmierbefehle | Befehls-mnemonik | Wort | | | Doppelwort | | |
|---|---|---|---|---|---|---|---|
| | | Ausführungszeit in µs | | Speicher-belegung in Worten | Ausführungszeit in µs | | Speicher-belegung in Worten |
| | | unwahr | wahr | | unwahr | wahr | |
| **Kleiner als oder gleich** | **LEQ** | 1,1 | 1,3 | 1,3 | 2,7 | 2,8 | 2,9 |
| **Kodieren 1 aus 16 in 4** | **ENC** | 0,0 | 7,2 | 1,5 | Keine Doppelwortadressierung. | | |
| **Kommunikationsbearbeitung** | **SVC** | 0,0 | 208 + 1,6/ Wort(2) | 1,0 | | | |
| **LIFO entladen** | **LFU** | 10,4 | 29,1 | 3,4 | 10,4 | 31,6 | 3,4 |
| **LIFO laden** | **LFL** | 10,4 | 25,5 | 3,4 | 10,4 | 31,6 | 3,9 |
| **Logisches NICHT** | **NOT** | 0,0 | 2,4 | 2,5 | 0,0 | 9,2 | 2,5 |
| **Logisches ODER** | **OR** | 0,0 | 2,2 | 2,8 | 0,0 | 9,2 | 3,0 |
| **Logisches UND** | **AND** | 0,0 | 2,2 | 2,8 | 0,0 | 9,2 | 3,0 |
| **Löschen** | **CLR** | 0,0 | 1,3 | 1,0 | 0,0 | 6,3 | 1,0 |
| **Marke** | **LBL** | 1,0 | 1,0 | 0,5 | Keine Doppelwortadressierung. | | |
| **Maskierter Vergleich auf Gleich** | **MEQ** | 1,8 | 1,9 | 1,8 | 3,1 | 3,9 | 3,5 |
| **Multiplikation** | **MUL** | 0,0 | 6,8 | 2,0 | 0,0 | 31,9 | 3,5 |
| **Nachricht lesen/schreiben, ständig** | **MSG** | 6,0 | 20,0 | 2,9 | Keine Doppelwortadressierung. | | |
| **Nachricht, Strompfadübergang von unwahr nach wahr für Lesenachrichten** | | | 230,0 | | | | |
| **Nachricht, Strompfadübergang von unwahr nach wahr für Schreibnachrichten** | | | 264 + 1,6/ Wort | | | | |
| **Negation** | **NEG** | 0,0 | 2,9 | 3,0 | 0,0 | 12,1 | 3,0 |
| **OSF - Fallender Einzelimpuls** | **OSF** | 3,7 | 2,8 | 5,4 | Keine Doppelwortadressierung. | | |
| **Proportional/Integral/ Differenzialverhalten** | **PID** | 11,0 | 295,8 | 2,4 | | | |
| **Pulsweitenmodulation(1)** | **PWM** | 24,7 | 126,6 | 1,9 | | | |
| **Quadratwurzel** | **SQR** | 0,0 | 26,0 | 1,5 | 0,0 | 30,9 | 2,5 |
| **Rücksetzen** | **RES** | 0,0 | 5,9 | 1,0 | Keine Doppelwortadressierung. | | |
| **Rücksprung** | **RET** | 0,0 | 1,0 | 0,3 | | | |
| **Rückwärtszählung** | **CTD** | 9,0 | 9,0 | 2,4 | | | |
| **Schrittschaltwerksausgang** | **SQO** | 7,1 | 23,2 | 3,9 | 7,1 | 26,6 | 4,4 |
| **Schrittschaltwerksvergleich** | **SQC** | 7,1 | 23,5 | 3,9 | 7,1 | 26,3 | 4,4 |
| **Schrittschaltwerksladung** | **SQL** | 7,0 | 21,7 | 3,4 | 7,1 | 24,3 | 3,9 |
| **Skalierung** | **SCL** | 0,0 | 10,5 | 2,5 | Keine Doppelwortadressierung. | | |
| **Skalierung mit Parametern** | **SCP** | 0,0 | 31,5 | 3,8 | 0,0 | 52,2 | 6,0 |
| **Sofortiger Ausgang mit Maske** | **IOM** | 0,0 | 22,3 | 3,0 | Keine Doppelwortadressierung. | | |
| **Sofortiger Eingang mit Maske** | **IIM** | 0,0 | 26,4 | 3,0 | | | |
| **Speicherndes Zeitwerk** | **RTO** | 2,4 | 18,0 | 3,4 | | | |
| **Sprung ins Unterprogramm** | **JSR** | 0,0 | 8,4 | 1,5 | | | |
| **Sprung zur Marke** | **JMP** | 0,0 | 1,0 | 0,5 | | | |
| **Steigender Einzelimpuls** | **OSR** | 3,0 | 3,4 | 5,4 | | | |
| **STI starten** | **STS** | 0,0 | 57,5 | 1,0 | | | |
| **Subtraktion** | **SUB** | 0,0 | 3,4 | 3,3 | 0,0 | 12,9 | 3,5 |
| **Suspend** | **SUS** | nicht vorhan-den | nicht vorhanden | 1,5 | Keine Doppelwortadressierung. | | |

**Tabelle A.1 Speicherbelegung und Ausführungszeiten der MicroLogix 1200-Programmierbefehle**

| Programmierbefehle | Befehls-mnemonik | Wort | | | Doppelwort | | |
|---|---|---|---|---|---|---|---|
| | | Ausführungszeit in µs | | Speicher-belegung in Worten | Ausführungszeit in µs | | Speicher-belegung in Worten |
| | | unwahr | wahr | | unwahr | wahr | |
| **Temporäres Ende** | **TND** | 0,0 | 0,9 | 0,5 | Keine Doppelwortadressierung. | | |
| **Timer-Ausschaltverzögerung** | **TOF** | 13,0 | 2,9 | 3,9 | | | |
| **Timer-Einschaltverzögerung** | **TON** | 3,0 | 18,0 | 3,9 | | | |
| **Übertragung** | **MOV** | 0,0 | 2,4 | 2,5 | 0,0 | 8,3 | 2,0 |
| **Übertragung durch Maske** | **MVM** | 0,0 | 7,8 | 2,0 | 0,0 | 11,8 | 3,0 |
| **Umwandlung von BCD** | **FRD** | 0,0 | 14,1 | 1,5 | Keine Doppelwortadressierung. | | |
| **Ungleich** | **NEQ** | 1,1 | 1,3 | 1,3 | 2,7 | 2,5 | 2,5 |
| **Unterprogramm** | **SBR** | 1,0 | 1,0 | 0,3 | Keine Doppelwortadressierung. | | |
| **Vorwärtszählung** | **CTU** | 9,2 | 9,0 | 2,4 | | | |
| **Wort kopieren** | **CPW** | 0,0 | 18,3 + 0,8/ Wort | | | | |

(1) Gilt nur für MicroLogix 1200 Steuerungen der Serie B.

(2) Dieser Wert für den Befehl SVC wird gesetzt, wenn der Kommunikationsdienst auf einen Datenfile zugreift. Die Zeit erhöht sich beim Zugreifen auf einen Funktionsfile.

## Indirekte Adressierung

In den folgenden Abschnitten werden die Auswirkungen der indirekten Adressierung auf die Ausführungszeit von Befehlen der Micrologix 1200-Steuerungen beschrieben. Diese Ausführungszeit wird beeinflusst von der Art der indirekten Adresse.

Für die Adressformate der nachfolgenden Tabelle sind folgende Filetypen gleich und damit austauschbar:

- Eingang (I) und Ausgang (O)
- Bit (B) und Ganzzahl (N)
- Zeitwerk (T), Zähler (C) und Steuerung (R)

### *Ausführungszeit für indirekte Adressen*

Für die meisten Befehlstypen, die indirekte Adressen enthalten, können die Art der Adressierung der folgenden Tabelle entnommen und die entsprechende Zeit zur Ausführungszeit des Befehls *addiert* werden.

[ * ] zeigt an, dass ein indirekter Bezugswert verwendet wurde.

**Tabelle A.2 Ausführungszeit von MicroLogix 1200-Befehlen bei indirekter Adressierung**

| Adressformat | Operandenzeit (µs) | Adressformat | Operandenzeit (µs) |
|---|---|---|---|
| O:1.[*] | 5,8 | B3:1/[*] | 6,8 |
| O:[*],0 | 15,0 | B3:[*]/[*] | 7,6 |
| O:[*],[*] | 15,1 | B[ * ]:1/[ * ] | 25,9 |
| B3:[*] | 5,8 | B[*]:[*]/[*] | 26,2 |
| B[*]:1 | 24,3 | L8:[*]/2 | 6,5 |
| B[*]:[*] | 24,5 | L[*]:1/2 | 24,6 |
| L8:[*] | 6,1 | L[*]:[*]/2 | 25,3 |
| L[*]:1 | 24,4 | L8:1/[*] | 6,8 |
| L[*]:[*] | 24,3 | L8:[*]/[*] | 7,7 |
| T4:[*] | 6,0 | L[*]:1/[*] | 26,0 |
| T[*]:1 | 24,0 | L[*]:[*]/[*] | 25,9 |
| T[*]:[*] | 24,2 | T4:[*]/DN | 6,6 |
| T4:[*],ACC | 6,5 | T[*]:1/DN | 24,4 |
| T[*]:1,ACC | 24,4 | T[*]:[*]/DN | 24,9 |
| T[*]:[*],ACC | 24,9 | T4:[*].ACC/2 | 7,4 |
| O:1,[*]/2 | 6,3 | T[*]:1,ACC/2 | 24,4 |
| O:[*],0/2 | 15,2 | T[*]:[*],ACC/2 | 25,9 |
| O:[*],[*]/2 | 15,9 | T4:1/[*] | 6,5 |
| O:1,0/[*] | 6,8 | T4:[*]/[*] | 8,3 |
| O:1,[*]/[*] | 7,6 | T[*]:1/[*] | 26,1 |
| O:[*],0/[*] | 16,6 | T[*]:[*]/[*] | 26,8 |
| O:[*],[*]/[*] | 16,9 | T4:1.ACC/[*] | 6,9 |
| B3:[*]/2 | 6,3 | T4:[*].ACC/[*] | 8,9 |
| B[*]:1/2 | 24,5 | T[*]:1,ACC/[*] | 26,1 |
| B[*]:[*]/2 | 25,3 | T[*]:[*],ACC/[*] | 27,3 |

*Beispiel Ausführungszeit – Wort-Befehl mit indirekter Adressierung*

**ADD-Befehlsadressierung**

- Quelle A: N7:[*]
- Quelle B: T4:[*],ACC
- Ziel: N[ * ]:[ * ]

**ADD-Ausführungszeiten**

- ADD-Befehl: 2,7 µs
- Quelle A: 5,8 µs
- Quelle B: 6,5 µs
- Ziel: 24,5 µs

Summe = 36,5 µs

*Beispiel Ausführungszeit – Bit-Befehl mit indirekter Adressierung*

**XIC B3/[*]**

- XIC: 0,9 µs + 5,8 µs = 6,7 µs bei wahr
- XIC: 0,9 µs + 5,8 µs = 6,7 µs bei unwahr

# MicroLogix 1200 – Arbeitsblatt zur Abfragezeit

Mit diesem Arbeitsblatt können Sie die Abfragezeit für ein Steuerprogramm berechnen.

| | | |
|---|---|---|
| **Eingangsabfrage (Summe der nachfolgenden Zwischensummen)** | | |
| Overhead (bei Verwendung von Erweiterungs-E/A) | = 55 µs | |
| Erweiterungs-Eingangsworte X 10 µs (oder X 14 µs bei Forcen) | = | |
| Anzahl Module mit Eingangsworten X 80 µs | = | |
| Eingangsabfrage Zwischensumme | | = |
| **Programmabfrage** | | |
| Addieren Sie die Ausführungszeiten aller Befehle in dem Programm bei wahrer Ausführung | = | |
| Programmabfrage Zwischensumme | | = |
| **Ausgangsabfrage (Summe der nachfolgenden Zwischensummen)** | | |
| Overhead (bei Verwendung von Erweiterungs-E/A) | = 30 µs | |
| Erweiterungs-Ausgangsworte X 3 µs (oder X 7 µs bei Forcen) | = | |
| Ausgangsabfrage Zwischensumme | | = |
| **Kommunikations-Overhead(1)** | | |
| Maximal | = 1470 µs | |
| Normalfall | = 530 µs | |
| Verwenden Sie diese Zahl, wenn der Kommunikationsanschluss konfiguriert ist, jedoch nicht mit anderen Geräten kommuniziert. | = 200 µs | |
| Verwenden Sie diese Zahl, wenn der Kommunikationsanschluss ausgeschaltet wurde. | = 0 µs | |
| Kommunikations-Overhead Zwischensumme | | = |
| **System-Overhead** | | |
| Addieren Sie diese Zahl, wenn das System ein 1762-RTC oder 1762-MM1RTC enthält. | = 100 µs | |
| Verwaltungs-Overhead | = 270 µs | |
| System-Overhead Zwischensumme | | = |
| **Summe** | | |
| Summe aller Zwischensummen | | |
| Multiplikation mit Kommunikationsmultiplikator aus Tabelle | | x |
| Geschätzte Abfragegesamtzeit | | = |

(1) Der Kommunikations-Overhead ist eine Funktion des Geräts, das an die Steuerung angeschlossen ist. Der Overhead tritt nicht bei jeder Abfrage auf.

## Tabelle Kommunikationsmultiplikator

| **Protokoll** | **Multiplikator bei verschiedenen Baudraten** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **38400** | **19200** | **9600** | **4800** | **2400** | **1200** | **600** | **300** | **Inaktiv(1)** |
| DF1Vollduplex | 1,50 | 1,27 | 1,16 | 1,12 | 1,10 | 1,09 | 1,09 | 1,08 | 1,00 |
| DF1 Halbduplex Slave | 1,21 | 1,14 | 1,10 | 1,09 | 1,08 | 1,08 | 1,08 | 1,07 | 1,01 |
| DH-485 | nicht zutreffend | 1,16 | 1,11 | nicht zutreffend | nicht zutreffend | nicht zutreffend | nicht zutreffend | nicht zutreffend | 1,10 bei 19200<br>1,07 bei 9600 |
| Modbus™ | 1,22 | 1,13 | 1,10 | 1,09 | 1,09 | 1,09 | 1,09 | 1,09 | 1,00 |
| ASCII | 1,55 | 1,33 | 1,26 | 1,22 | 1,21 | 1,19 | 1,19 | 1,18 | 1,01 |
| Abschalten | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 |

(1) Inaktiv ist definiert als Zustand ohne Nachrichtenübertragung und Datenüberwachung. Beim DH-485-Protokoll bedeutet „inaktiv“, dass die Steuerung nicht an ein Netzwerk angeschlossen ist.

# MicroLogix 1500 – Speicherbelegung und Befehlsausführungszeit

Dieser Anhang enthält eine vollständige Liste der MicroLogix 1500-Programmierbefehle. In dieser Liste sind die Speicherbelegung und die Ausführungszeit für jeden Befehl aufgeführt. Außerdem wird die Ausführungszeit bei indirekter Adressierung angegeben und ein Arbeitsblatt zur Berechnung der Abfragezeit bereitgestellt.

## Speicherbelegung und Ausführungszeit von Programmierbefehlen

Die folgenden Tabellen zeigen Ausführungsdauer und Speicherbedarf der einzelnen Steuerungsbefehle. Diese Werte sind abhängig davon, ob als Datenformat *Worte* oder *Doppelworte* verwendet werden.

**Tabelle B.1 MicroLogix 1500-Steuerungen – Speicherbelegung und Ausführungszeiten von Programmierbefehlen**

| Programmierbefehle | Befehls-mnemonik | Wort | | | Doppelwort | | |
|---|---|---|---|---|---|---|---|
| | | Ausführungszeit in µs | | Speicher-belegung in Worten | Ausführungszeit in µs | | Speicher-belegung in Worten |
| | | unwahr | wahr | | unwahr | wahr | |
| **4 in 1 aus 16 dekodieren** | **DCD** | 0,0 | 0,9 | 1,9 | Keine Doppelwortadressierung. | | |
| **Absolutwert** | **ABS** | 0,0 | 3,1 | | | | |
| **Addition** | **ADD** | 0,0 | 2,5 | 3,3 | 0,0 | 10,4 | 3,5 |
| **Anwender-Interrupt aktivieren** | **UIE** | 0,0 | 0,8 | 0,9 | Keine Doppelwortadressierung. | | |
| **Anwender-Interrupt deaktivieren** | **UID** | 0,0 | 0,8 | 0,9 | | | |
| **Anwender-Interrupt entfernen** | **UIF** | 0,0 | 10,6 | 0,9 | | | |
| **Anzahl der ASCII-Zeichen im Puffer(1)** | **ACB** | 11,0 | 84,2 | 3,3 | | | |
| **ASCII schreiben und anhängen(1)** | **AWA** | 12,5 | 236 + 10,6/ Zeichen | 3,4 | | | |
| **ASCII schreiben(1)** | **AWT** | 12,8 | 237 + 10,6/ Zeichen | 3,4 | | | |
| **ASCII-Ganzzahl in Zeichenkette(1)** | **AIC** | 0,0 | 25 + 4,3/Zeichen | 1,4 | 0,0 | 68,7 | 1,6 |
| **ASCII-Handshake-Leitungen(1)** | **AHL** | 10,8 | 89,3 | 5,3 | Keine Doppelwortadressierung. | | |
| **ASCII-Lesen(1)** | **ARD** | 10,7 | 108 + 44/Zeichen | 4,3 | | | |
| **ASCII-Pufferspeicher löschen(1)** | **ACL** | 0,0 | löschen: beide 203,9 empfangen 24,7 übertragen 29,1 | 1,2 | | | |
| **ASCII-Testpuffer für Zeile(1)** | **ABL** | 11,4 | 94 + 7,6/Zeichen | 3,3 | | | |
| **ASCII-Zeichenkette extrahieren(1)** | **AEX** | 0,0 | 12,4 + 2,6/ Zeichen | 2,5 | | | |
| **ASCII-Zeichenkette in Ganzzahl(1)** | **ACI** | 0,0 | 14,2 + 6,3/ Zeichen | 1,5 | 0,0 | 20,3 + 9,5/ Zeichen | 1,5 |

**Tabelle B.1 MicroLogix 1500-Steuerungen –
Speicherbelegung und Ausführungszeiten von Programmierbefehlen**

| Programmierbefehle | Befehls-mnemonik | Wort | | | Doppelwort | | |
|---|---|---|---|---|---|---|---|
| | | Ausführungszeit in µs | | Speicher-belegung in Worten | Ausführungszeit in µs | | Speicher-belegung in Worten |
| | | unwahr | wahr | | unwahr | wahr | |
| **ASCII-Zeichenkette suchen[1]** | **ASC** | 0,0 | 13,4 + 3,5/ übereinst. Zeichen | 6,0 | Keine Doppelwortadressierung. | | |
| **ASCII-Zeichenkette vergleichen[1]** | **ASR** | 0,0 | 7,5 + 3,5/ übereinst. Zeichen | 1,8 | | | |
| **ASCII-Zeichenkette verketten[1]** | **ACN** | 0,0 | 17,9 + 10,2/ Zeichen | 2,0 | | | |
| **ASCII-Zeile lesen[1]** | **ARL** | 10,6 | 114 + 44,3/ Zeichen | 4,3 | | | |
| **Auf geschlossen prüfen** | **XIC** | 0,0 | 0,9 | 1,0 | | | |
| **Auf offen prüfen** | **XIO** | 0,0 | 0,9 | 1,0 | | | |
| **Ausgang einschalten** | **OTE** | 0,0 | 1,2 | 1,6 | | | |
| **Ausgang rücksetzen** | **OTU** | 0,0 | 0,9 | 0,6 | | | |
| **Ausgang setzen** | **OTL** | 0,0 | 0,9 | 0,6 | | | |
| **Bit nach links verschieben** | **BSL** | 1,4 | 26,4 + 1,06/Wort | 3,8 | | | |
| **Bit nach rechts verschieben** | **BSR** | 1,4 | 26,1 + 1,07/Wort | 3,8 | | | |
| **Byte-Tausch[1]** | **SWP** | 0,0 | 11,7 + 1,8/ vertauschtem Wort | 1,5 | | | |
| **Data Log (Datenprotokoll)** | **DLG** | 6,7 | 67,5 + 11,8/ Datumsstempel +12,4/ Zeitstempel +9,1/ protokolliertes Wort | 2,4 | 6,7 | 67,5 + 11,8/ Datumsstempel +12,4/ Zeitstempel +16,2/ protokolliertes Doppelwort | 2,4 |
| **Division** | **DIV** | 0,0 | 10,3 | 2,0 | 0,0 | 36,7 | 3,5 |
| **E/AAuffrischung** | **REF** | 0,0 | siehe S. B-7 | 0,5 | Keine Doppelwortadressierung. | | |
| **Echtzeituhr anpassen** | **RTA** | 2,6 | 4,1 (426,8 unwahr-zu-wahr-Wechsel) | | | | |
| **Einschaltverzögerung** | **TON** | 2,5 | 15,5 | 3,9 | | | |
| **Einzelimpuls** | **ONS** | 1,7 | 2,2 | 3,5 | | | |
| **ExklusivOder** | **XOR** | 0,0 | 2,3 | 2,8 | 0,0 | 8,9 | 3,0 |
| **FIFO entladen** | **FFU** | 9,7 | 27,7 + 0,65/Wort | 3,4 | 9,7 | 29,4 + 1,25/ Doppelwort | 3,4 |
| **FIFO laden** | **FFL** | 9,8 | 10,0 | 3,4 | 9,7 | 10,9 | 3,9 |
| **File kopieren** | **COP** | 0,0 | 15,9 + 0,67/Wort | 2,0 | Keine Doppelwortadressierung. | | |
| **Filefüllung** | **FLL** | 0,0 | 12,1 + 0,43/Wort | 2,0 | 0,0 | 12,3 + 0,8/ Doppelwort | 2,5 |
| **Gleich** | **EQU** | 1,1 | 1,2 | 1,3 | 1,9 | 2,6 | 2,6 |
| **Gray-Code** | **GCD** | 0,0 | 9,5 | | Keine Doppelwortadressierung. | | |
| **Grenzwerttest** | **LIM** | 5,3 | 5,5 | 2,3 | 11,7 | 12,2 | 4,0 |
| **Größer als** | **GRT** | 1,1 | 1,2 | 1,3 | 2,5 | 2,6 | 2,4 |
| **Größer als oder gleich** | **GEQ** | 1,1 | 1,2 | 1,3 | 2,5 | 2,6 | 2,9 |

**Tabelle B.1 MicroLogix 1500-Steuerungen –
Speicherbelegung und Ausführungszeiten von Programmierbefehlen**

| Programmierbefehle | Befehls-mnemonik | Wort | | | Doppelwort | | |
|---|---|---|---|---|---|---|---|
| | | Ausführungszeit in µs | | Speicher-belegung in Worten | Ausführungszeit in µs | | Speicher-belegung in Worten |
| | | unwahr | wahr | | unwahr | wahr | |
| **Hauptsteuerbefehl** | **MCR (Start)** | 0,8 | 0,8 | 1,0 | Keine Doppelwortadressierung. | | |
| | **MCR (Ende)** | 1,0 | 1,0 | 1,5 | | | |
| **Hochgeschwindigkeitsladen** | **HSL** | 0,0 | 39,7 | 7,3 | 0,0 | 40,3 | 7,8 |
| **Impulsgang** | **PTO** | 21,1 | 72,6 | 1,9 | Keine Doppelwortadressierung. | | |
| **In BCD** | **TOD** | 0,0 | 14,3 | 1,8 | | | |
| **Interrupt Subroutine** | **INT** | 1,0 | 1,0 | 0,3 | | | |
| **Istwert rücksetzen** | **RAC** | Keine Wortadressierung. | | | 0,0 | 17,8 | 2,0 |
| **Kleiner als** | **LES** | 1,1 | 1,2 | 1,3 | 2,5 | 2,6 | 2,9 |
| **Kleiner als oder gleich** | **LEQ** | 1,1 | 1,2 | 1,3 | 2,5 | 2,6 | 2,9 |
| **Kodieren 1 aus 16 in 4** | **ENC** | 0,0 | 6,8 | 1,5 | Keine Doppelwortadressierung. | | |
| **Kommunikationsbearbeitung (Bearbeitung eines Kanals)** | **SVC(2)** | 0,0 | 166 + 1,4/Wort | 1,0 | | | |
| **Kommunikationsbearbeitung (Bearbeitung zweier Kanäle)** | | 0,0 | 327 + 1,4/Wort | 1,0 | | | |
| **LIFO entladen** | **LFU** | 9,7 | 25,6 | 3,4 | 9,7 | 27,4 | 3,4 |
| **LIFO laden** | **LFL** | 9,7 | 22,2 | 3,4 | 9,7 | 27,4 | 3,9 |
| **Logisches NICHT** | **NOT** | 0,0 | 2,4 | 2,5 | 0,0 | 8,1 | 2,5 |
| **Logisches ODER** | **OR** | 0,0 | 2,0 | 2,8 | 0,0 | 7,9 | 3,0 |
| **Logisches UND** | **AND** | 0,0 | 2,0 | 2,8 | 0,0 | 7,9 | 3,0 |
| **Löschen** | **CLR** | 0,0 | 1,2 | 1,0 | 0,0 | 5,5 | 1,0 |
| **Marke** | **LBL** | 1,0 | 1,0 | 0,5 | Keine Doppelwortadressierung. | | |
| **Maskierter Vergleich auf Gleich** | **MEQ** | 1,7 | 1,7 | 1,8 | 2,9 | 3,5 | 3,5 |
| **Multiplikation** | **MUL** | 0,0 | 5,8 | 2,0 | 0,1 | 27,6 | 3,5 |
| **Nachricht lesen/schreiben, ständig** | **MSG** | 6,0 | 17,0 | 2,9 | Keine Doppelwortadressierung. | | |
| **Nachricht, Strompfadübergang von unwahr nach wahr für Lesenachrichten** | | | 198,0 | | | | |
| **Nachricht, Strompfadübergang von unwahr nach wahr für Schreibnachrichten** | | | 226 + 1,4/Wort | | | | |
| **Negation** | **NEG** | 0,0 | 1,9 | 3,0 | 0,0 | 10,4 | 3,0 |
| **OSF - Fallender Einzelimpuls** | **OSF** | 3,4 | 2,7 | 5,4 | Keine Doppelwortadressierung. | | |
| **Proportional/Integral/ Differenzialverhalten** | **PID** | 8,9 | 251,8 | 2,4 | | | |
| **Pulsdauermodulation** | **PWM** | 21,1 | 107,4 | 1,9 | | | |
| **Quadratwurzel** | **SQR** | 0,0 | 22,3 | 1,5 | 0,0 | 26,0 | 2,5 |
| **Rücksetzen** | **RES** | 0,0 | 4,8 | 1,0 | Keine Doppelwortadressierung. | | |
| **Rücksprung** | **RET** | 0,0 | 1,0 | 0,3 | | | |
| **Rückwärtszählung** | **CTD** | 8,5 | 7,5 | 2,4 | | | |
| **Schrittschaltwerks ausgang** | **SQO** | 6,3 | 20,0 | 3,9 | 6,3 | 23,1 | 4,4 |
| **Schrittschaltwerks vergleich** | **SQC** | 6,3 | 20,1 | 3,9 | 6,3 | 22,7 | 4,4 |
| **Schrittschaltwerksladung** | **SQL** | 6,3 | 19,1 | 3,4 | 6,3 | 21,1 | 3,9 |
| **Skalierung** | **SCL** | 0,0 | 8,7 | 2,5 | Keine Doppelwortadressierung. | | |

**Tabelle B.1 MicroLogix 1500-Steuerungen –
Speicherbelegung und Ausführungszeiten von Programmierbefehlen**

| Programmierbefehle | Befehls-mnemonik | Wort | | | Doppelwort | | |
|---|---|---|---|---|---|---|---|
| | | Ausführungszeit in µs | | Speicher-belegung in Worten | Ausführungszeit in µs | | Speicher-belegung in Worten |
| | | unwahr | wahr | | unwahr | wahr | |
| **Skalierung mit Parametern** | **SCP** | 0,0 | 27,0 | 3,8 | 0,0 | 44,7 | 6,0 |
| **Sofortiger Ausgang mit Maske** | **IOM** | 0,0 | 19,4 | 3,0 | Keine Doppelwortadressierung. | | |
| **Sofortiger Eingang mit Maske** | **IIM** | 0,0 | 22,5 | 3,0 | | | |
| **Speicherndes Zeitwerk** | **RTO** | 2,2 | 15,8 | 3,4 | | | |
| **Sprung ins Unterprogramm** | **JSR** | 0,0 | 8,0 | 1,5 | | | |
| **Sprung zur Marke** | **JMP** | 0,0 | 1,0 | 0,5 | | | |
| **Steigender Einzelimpuls** | **OSR** | 2,8 | 3,2 | 5,4 | | | |
| **STI starten** | **STS** | 0,0 | 50,7 | 1,0 | Keine Doppelwortadressierung. | | |
| **Subtraktion** | **SUB** | 0,0 | 2,9 | 3,3 | 0,0 | 11,2 | 3,5 |
| **Suspend** | **SUS** | nicht zutreffend | nicht zutreffend | 1,5 | Keine Doppelwortadressierung. | | |
| **Temporäres Ende** | **TND** | 0,0 | 1,0 | 0,5 | | | |
| **Timer-Ausschaltverzögerung** | **TOF** | 10,9 | 2,5 | 3,9 | | | |
| **Übertragung** | **MOV** | 0,0 | 2,3 | 2,5 | 0,0 | 6,8 | 2,0 |
| **Übertragung durch Maske** | **MVM** | 0,0 | 7,2 | 2,0 | 0,0 | 10,0 | 3,0 |
| **Umwandlung von BCD** | **FRD** | 0,0 | 12,3 | 1,5 | Keine Doppelwortadressierung. | | |
| **Ungleich** | **NEQ** | 1,1 | 1,2 | 1,3 | 2,5 | 2,3 | 2,5 |
| **Unterprogramm** | **SBR** | 1,0 | 1,0 | 0,3 | Keine Doppelwortadressierung. | | |
| **Vorwärtszählung** | **CTU** | 8,5 | 6,4 | 2,4 | | | |
| **Wort kopieren** | **CPW** | 0,0 | 15,8 + 0,7/Wort | | | | |
| **Anwender-Interrupt aktivieren** | **UIE** | 0,0 | 0,8 | 0,9 | Keine Doppelwortadressierung. | | |
| **Anwender-Interrupt deaktivieren** | **UID** | 0,0 | 0,8 | 0,9 | | | |
| **Anwender-Interrupt entfernen** | **UIF** | 0,0 | 10,6 | 0,9 | | | |
| **Anzahl der ASCII-Zeichen im Puffer(1)** | **ACB** | 11,0 | 84,2 | 3,3 | | | |
| **ASCII schreiben und anhängen(1)** | **AWA** | 12,5 | 236 + 10,6/ Zeichen | 3,4 | | | |
| **ASCII schreiben(1)** | **AWT** | 12,8 | 237 + 10,6/ Zeichen | 3,4 | | | |
| **ASCII-Ganzzahl in Zeichenkette(1)** | **AIC** | 0,0 | 25 + 4,3/Zeichen | 1,4 | 0,0 | 68,7 | 1,6 |

(1) Gilt nur für MicroLogix 1500-Prozessoren der Serie B.

(2) Dieser Wert für den Befehl SVC wird gesetzt, wenn der Kommunikationsdienst auf einen Datenfile zugreift. Die Zeit erhöht sich beim Zugreifen auf einen Funktionsfile.

## Indirekte Adressierung

In den folgenden Abschnitten werden die Auswirkungen der indirekten Adressierung auf die Ausführungszeit von Befehlen in dem Micrologix 1500-Prozessor beschrieben. Diese Ausführungszeit wird beeinflusst von der Art der indirekten Adresse.

Für die Adressformate der nachfolgenden Tabelle sind folgende Filetypen gleich und damit austauschbar:

- Eingang (I) und Ausgang (O)
- Bit (B) und Ganzzahl (N)
- Zeitwerk (T), Zähler (C) und Steuerung (R)

### *Ausführungszeit für indirekte Adressen*

Für die meisten Befehlstypen, die indirekte Adressen enthalten, können die Art der Adressierung der folgenden Tabelle entnommen und die entsprechende Zeit zur Ausführungszeit des Befehls **addiert** werden.

[ * ] zeigt an, dass ein indirekter Bezugswert verwendet wurde.

**Tabelle B.2 MicroLogix 1500-Steuerungen – Befehlsausführungszeit bei indirekter Adressierung**

| Adress-format | Operanden-zeit (µs) | Adress-format | Operanden-zeit (µs) | Adress-format | Operanden-zeit (µs) |
|---|---|---|---|---|---|
| O:1.[*] | 4,8 | O:[*],[*]/2 | 13,3 | L[*]:1/[*] | 21,6 |
| O:[*],0 | 12,3 | O:1,0/[*] | 5,9 | L[*]:[*]/[*] | 21,9 |
| O:[*],[*] | 12,4 | O:1,[*]/[*] | 6,5 | T4:[*]/DN | 5,7 |
| B3:[*] | 4,8 | O:[*],0/[*] | 14,1 | T[*]:1/DN | 20,4 |
| B[*]:1 | 19,9 | O:[*],[*]/[*] | 14,5 | T[*]:[*]/DN | 20,7 |
| B[*]:[*] | 20,1 | B3:[*]/2 | 5,4 | T4:[*].ACC/2 | 6,4 |
| L8:[*] | 5,2 | B[*]:1/2 | 20,4 | T[*]:1,ACC/2 | 20,4 |
| L[*]:1 | 20,4 | B[*]:[*]/2 | 21,0 | T[*]:[*],ACC/2 | 21,6 |
| L[*]:[*] | 20,1 | B3:1/[*] | 5,9 | T4:1/[*] | 5,9 |
| T4:[*] | 4,9 | B3:[*]/[*] | 6,5 | T4:[*]/[*] | 7,1 |
| T[*]:1 | 19,7 | B[ * ]:1/[ * ] | 21,6 | T[*]:1/[*] | 21,8 |
| T[*]:[*] | 19,8 | B[*]:[*]/[*] | 22,3 | T[*]:[*]/[*] | 22,4 |
| T4:[*],ACC | 5,1 | L8:[*]/2 | 5,5 | T4:1.ACC/[*] | 6,0 |
| T[*]:1,ACC | 19,9 | L[*]:1/2 | 20,4 | T4:[*].ACC/[*] | 7,5 |
| T[*]:[*],ACC | 20,5 | L[*]:[*]/2 | 21,0 | T[*]:1,ACC/[*] | 21,8 |
| O:1,[*]/2 | 5,4 | L8:1/[*] | 5,9 | T[*]:[*],ACC/[*] | 22,9 |
| O:[*],0/2 | 12,8 | L8:[*]/[*] | 6,5 | | |

*Beispiel Ausführungszeit – Wort-Befehl mit indirekter Adressierung*

| ADD-Befehlsadressierung | ADD-Ausführungszeiten |
|---|---|
| | ADD-Befehl: 2,5 µs |
| Quelle A: N7:[*] | Quelle A: 4,8 µs |
| Quelle B: T4:[*],ACC | Quelle B: 5,1 µs |
| Ziel: N[ * ]:[ * ] | Ziel: 20,1 µs |
| | **Summe = 32,5 µs** |

*Beispiel Ausführungszeit – Bit-Befehl mit indirekter Adressierung*

**XIC B3/[*]**

- XIC: 0,9 µs + 4,8 µs = 5,7 µs bei wahr
- XIC: 0,0 µs + 4,8 µs = 4,8 µs bei unwahr

# MicroLogix 1500 – Arbeitsblatt zur Abfragezeit

Mit diesem Arbeitsblatt können Sie die Abfragezeit für ein Steuerprogramm berechnen.

| | | |
|---|---|---|
| **Eingangsabfrage (Summe der nachfolgenden Zwischensummen)** | | |
| Overhead (bei Verwendung von Erweiterungs-E/A) | = 53 µs | |
| Erweiterungs-Eingangsworte X 3 µs (oder X 7,5 µs bei Forcen) | = | |
| Anzahl Module mit Eingangsworten X 10 µs | = | |
| Eingangsabfrage Zwischensumme | | = |
| **Programmabfrage** | | |
| Addieren Sie die Ausführungszeiten aller Befehle in dem Programm bei wahrer Ausführung. | = | |
| Programmabfrage Zwischensumme | | = |
| **Ausgangsabfrage (Summe der nachfolgenden Zwischensummen)** | | |
| Overhead (bei Verwendung von Erweiterungs-E/A) | = 29 µs | |
| Erweiterungs-Ausgangsworte X 2 µs (oder X 6,5 µs bei Forcen) | = | |
| Ausgangsabfrage Zwischensumme | | = |
| **Kommunikations-Overhead(1)** | | |
| Maximal | = 1100 µs | |
| Normalfall | = 400 µs | |
| Verwenden Sie diese Zahl, wenn der Kommunikationsanschluss konfiguriert ist, jedoch nicht mit anderen Geräten kommuniziert. | = 150 µs | |
| Verwenden Sie diese Zahl, wenn der Kommunikationsanschluss ausgeschaltet wurde. | = 0 µs | |
| Wählen Sie eine der vier Zahlen für Kanal 0 | | |
| Wählen Sie eine der vier Zahlen für Kanal 1 | | |
| Kommunikations-Overhead Zwischensumme | | = |
| **System-Overhead** | | |
| Addieren Sie diese Zahl, wenn das System ein 1764-RTC-, 1764-MM1RTC- oder MM2RTC-Modul enthält. | = 80 µs | |
| Addieren Sie diese Zahl, wenn das System ein 1764-DAT enthält. | = 530 µs | |
| Verwaltungs-Overhead | = 240 µs | 240 |
| System-Overhead Zwischensumme | | = |
| **Summe** | | |
| Gesamtsumme | | |
| Multiplikation mit Kommunikationsmultiplikator aus Tabelle | | x |
| Multiplikator für Zeitsteuertakt (X1,02) | | |
| Geschätzte Abfragegesamtzeit | | = |

(1) Der Kommunikations-Overhead ist eine Funktion des Geräts, das an die Steuerung angeschlossen ist. Der Overhead tritt nicht bei jeder Abfrage auf.

## Tabelle Kommunikationsmultiplikator

| | **Multiplikator bei verschiedenen Baudraten** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Protokoll** | **38400** | **19200** | **9600** | **4800** | **2400** | **1200** | **600** | **300** | **Inaktiv(1)** |
| DF1Vollduplex | 1,39 | 1,20 | 1,13 | 1,10 | 1,09 | 1,08 | 1,08 | 1,08 | 1,00 |
| DF1-Halbduplex | 1,18 | 1,12 | 1,09 | 1,08 | 1,07 | 1,07 | 1,06 | 1,06 | 1,01 |
| DH-485 | nicht zutreffend | 1,14 | 1,10 | nicht zutreffend | nicht zutreffend | nicht zutreffend | nicht zutreffend | nicht zutreffend | 1,06 bei 19200<br>1,09 bei 9600 |
| Modbus(2) | 1,21 | 1,12 | 1,09 | 1,08 | 1,08 | 1,08 | 1,08 | 1,08 | 1,00 |
| ASCII(2) | 1,52 | 1,33 | 1,24 | 1,20 | 1,19 | 1,18 | 1,18 | 1,17 | 1,00 |
| Abschalten | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 | 1,00 |

(1) Inaktiv ist definiert als Zustand ohne Nachrichtenübertragung und Datenüberwachung. Beim DH-485-Protokoll bedeutet „inaktiv", dass die Steuerung nicht an ein Netzwerk angeschlossen ist.

(2) Gilt nur für MicroLogix 1500-Prozessoren der Serie B.

# Systemstatusfile

Mit dem Statusfile können Sie die Aktivität der Steuerung überwachen und beeinflussen. Dabei werden mit Hilfe des Statusfiles Steuer-Bits eingerichtet und Hardware- und Programmiergerätefehler sowie andere Statusinformationen überwacht.

**WICHTIG** In die reservierten Worte im Statusfile darf nicht geschrieben werden. Bevor Sie in den Statusfile schreiben, sollten Sie sich vollständig mit dessen Funktion vertraut machen.

# Übersicht Statusfile

Der Statusfile (S:) enthält die folgenden Worte:

| Adresse | Funktion | Seite |
|---|---|---|
| S:0 | Arithmetik-Flags | C-3 |
| S:1 | Steuerungsmodus | C-4 |
| S:2 | STI-Modus | C-10 |
| S:2/9 | Speichermodul-Programmvergleich | C-10 |
| S:2/15 | Auswahl der mathematischen Überlauffunktion | C-11 |
| S:3H | Watchdog-Abfragezeit | C-11 |
| S:4 | Freilaufender Takt | C-12 |
| S:5 | Bits für geringfügige Fehler | C-12 |
| S:6 | Fehlercode für schwerwiegende Fehler | C-14 |
| S:7 | Suspend-Code | C-15 |
| S:8 | Suspend-File | C-15 |
| S:9 | Aktive Netznoten (Netzknoten 0 bis 15) | C-15 |
| S:10 | Aktive Netzknoten (Netzknoten 16 bis 31) | C-15 |
| S:13, S:14 | Rechenregister | C-16 |
| S:15L | Netzknotenadresse | C-16 |
| S:15H | Baudrate | C-16 |
| S:22 | Maximale Scanzeit | C-16 |
| S:29 | Filenummer für Benutzerfehler-Routine | C-17 |
| S:30 | STI-Sollwert | C-17 |
| S:31 | STI-Filenummer | C-17 |
| S:33 | Kommunikation Kanal 0 | C-17 |
| S:35 | Letzte 100 µs Scanzeit | C-18 |
| S:36/10 | Datenfile-Überschreibschutz inaktiv | C-19 |
| S:37 | RTC-Jahr | C-19 |
| S:38 | RTC-Monat | C-19 |
| S:39 | RTC-Tag | C-19 |
| S:40 | RTC-Stunden | C-20 |
| S:41 | RTC-Minuten | C-20 |
| S:42 | RTC-Sekunden | C-20 |
| S:53 | RTC-Wochentag | C-20 |
| S:57 | Bestellnummer Betriebssystem | C-20 |
| S:58 | Betriebssytemserie | C-21 |
| S:59 | Betriebssytem-FRN | C-21 |
| S:60 | Prozessor-Bestellnummer | C-21 |
| S:61 | Prozessorserie | C-21 |
| S:62 | Prozessorversion | C-21 |
| S:63 | Benutzerprogramm-Funktionstyp | C-21 |
| S:64L | Compiler-Revision – Build-Nummer | C-21 |
| S:64H | Compiler-Revision – Versionsnummer | C-22 |

# Details des Statusfiles

## Arithmetik-Flags

Die Arithmetik-Flags werden vom Prozessor im Anschluss an einen Rechen-, Logik- oder Übertragungsbefehl ausgewertet. Der Status dieser Bits bleibt bis zur Ausführung des nächsten Rechen-, Logik- bzw. Übertragungsbefehls unverändert.

### *Übertrag-Flag*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:0/0 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt (1), wenn ein mathematischer Übertrag oder Abzug generiert wird. Andernfalls ist das Bit nicht gesetzt (0). Wenn ein STI-Befehl, ein Hochgeschwindigkeitszähler, ein Ereignis-Interrupt oder eine Anwenderfehlerroutine die normale Ausführung des Pro- gramms unterbricht, wird der ursprüngliche Wert von S:0/0 bei der Fortsetzung der Programmausführung wiederhergestellt.

### *Überlauf-Flag*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:0/1 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt (1), wenn im Ziel kein ausreichender Speicher- platz für das Ergebnis eines mathematischen Befehls vorhanden ist. Andernfalls ist das Bit nicht gesetzt (0). Sobald dieses Bit gesetzt wird (1), wird auch das Überlauferkennungs-Bit S:5/0 gesetzt (1). Wenn ein STI-Befehl, ein Hochgeschwindigkeitszähler, ein Ereignis-Interrupt oder eine Anwenderfehlerroutine die normale Ausführung des Pro- gramms unterbricht, wird der ursprüngliche Wert von S:0/1 bei der Fortsetzung der Programmausführung wiederhergestellt.

### *Null-Flag*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:0/2 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt (1), wenn das Ergebnis eines Mathematik- oder Datenverarbeitungsbefehls gleich null ist. Andernfalls ist das Bit nicht gesetzt (0). Wenn ein STI-Befehl, ein Hochgeschwindigkeitszähler, ein Ereignis-Interrupt oder eine Anwenderfehlerroutine die normale Aus- führung des Programms unterbricht, wird der ursprüngliche Wert von S:0/2 bei der Fortsetzung der Programmausführung wiederhergestellt.

*Zeichen-Flag*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:0/3 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt (1), wenn das Ergebnis eines Mathematik- oder Datenverarbeitungsbefehls negativ ist. Andernfalls ist das Bit nicht gesetzt (0). Wenn ein STI-Befehl, ein Hochgeschwindigkeitszähler, ein Ereignis-Interrupt oder eine Anwenderfehlerroutine die normale Aus- führung des Programms unterbricht, wird der ursprüngliche Wert von S:0/3 bei der Fortsetzung der Programmausführung wiederhergestellt.

## Steuerungsmodus

*Benutzeranwendungsmodus*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/0 bis S:1/4 | Binärwert | 0 bis 1 1110 | Status | Nur Lesen |

Funktionsweise der Bits 0 bis 4:

| S:1/0 bis S:1/4 | | | | | Modus-ID | Steuerungsmodus | Verwendung durch MicroLogix-Steuerung(1) | |
|---|---|---|---|---|---|---|---|---|
| S:1/4 | S:1/3 | S:1/2 | S:1/1 | S:1/0 | | | 1200 | 1500 |
| 0 | 0 | 0 | 0 | 0 | 0 | dezentraler Herunterladevorgang läuft | • | • |
| 0 | 0 | 0 | 0 | 1 | 1 | REM Programmmodus | • | • |
| 0 | 0 | 0 | 1 | 1 | 3 | dezentraler Suspend-Zustand (Operation durch Ausführung des SUS-Befehls angehalten) | • | • |
| 0 | 0 | 1 | 1 | 0 | 6 | dezentraler RunModus | • | • |
| 0 | 0 | 1 | 1 | 1 | 7 | dezentraler Dauertestmodus | • | • |
| 0 | 1 | 0 | 0 | 0 | 8 | dezentraler Einzelabfrage-Testmodus | • | • |
| 1 | 0 | 0 | 0 | 0 | 16 | Herunterladevorgang läuft | nicht zutreffend | • |
| 1 | 0 | 0 | 0 | 1 | 17 | Programmmodus | nicht zutreffend | • |
| 1 | 1 | 0 | 1 | 1 | 27 | Suspend-Zustand (Operation durch Ausführung des SUS-Befehls angehalten) | nicht zutreffend | • |
| 1 | 1 | 1 | 1 | 0 | 30 | Run-Modus | nicht zutreffend | • |

(1) Gültige Modi sind durch das Symbol (•) gekennzeichnet. N/A steht für einen ungültigen Modus für die betreffende Steuerung.

*Forcen aktiv*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/5 | Binärwert | 1 | Status | Nur Lesen |

Dieses Bit wird von der Steuerung immer gesetzt (1), um anzuzeigen, dass Forcen aktiviert ist.

### *Forcen installiert*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/6 | Binärwert | 0 oder 1 | Status | Nur Lesen |

Dieses Bit wird von der Steuerung gesetzt (1), wenn ein oder mehrere Ein- oder Ausgänge forciert werden. Wenn dieses Bit nicht gesetzt ist, ist in der Steuerung Forcen nicht aktiviert.

### *Fehlerüberbrückung beim Einschalten*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/8 | Binärwert | 0 oder 1 | Steuerung | Nur Lesen |

Wenn dieses Bit gesetzt ist (1), wird das Fehler-Bit (S:1/13) (Halt wegen schwerem Fehler) beim Einschalten gelöscht. Der Einschaltmodus wird durch den Steuerungsmodusschalter *(nur MicroLogix 1500)* und das Auswahl-Bit für das Einschaltmodus- verhalten (S:1/12) bestimmt.

Siehe auch: „FO – Fehler überbrücken“ auf Seite 3-8.

### *Anlauffehlerschutz*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/9 | Binärwert | 0 oder 1 | Steuerung | Nur Lesen |

Wenn dieses Bit gesetzt ist (1) und sich der Prozessor im RUN- oder REM Run-Modus befindet, führt der Prozessor vor der ersten Programmabfrage die Anwenderfehlerroutine aus. Dabei haben Sie die Möglichkeit, das Fehler-Bit (S:1/13) (Halt wegen schwerem Fehler) zu löschen, um den normalen Betrieb wiederaufzunehmen. Wenn das Bit S:1/13 nicht von der Anwenderfehlerroutine gelöscht wird, tritt ein Steuerungsfehler auf und der Ausführungsmodus der Steuerung wird nicht aktiviert. Programmieren Sie die Logik der Anwenderfehler- routine dementsprechend.

**HINWEIS** Bei der Ausführung der Anlauffehlerschutzroutine enthält das Bit S:6 (schwerwiegender Fehlercode) den Wert 0016H.

### *Speichermodul bei Fehler oder Standardprogramm laden*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/10 | Binärwert | 0 oder 1 | Steuerung | Nur Lesen |

Diese Funktion ist nur verfügbar, wenn dieses Bit im Steuerprogramm gesetzt wird (1), bevor das Programm auf das Speichermodul herun- tergeladen wird. Wenn dieses Bit im Speichermodul gesetzt ist und die Spannungsversorgung eingeschaltet wird, lädt die Steuerung das Speichermodulprogramm herunter, wenn das Steuerprogramm fehlerhaft ist oder ein Standardprogramm in der Steuerung vorhanden ist.

**HINWEIS** Wenn Sie den Speicher der Steuerung löschen, lädt die Steuerung das Standardprogramm.

Der Steuerungsmodus nach der Übertragung wird durch den Steuerungsmodusschalter *(nur MicroLogix 1500)* und das Auswahl-Bit für das Einschaltmodusverhalten (S:1/12) bestimmt.

Siehe auch:„LE – Bei Fehler laden“ auf Seite 3-8.

### *Speichermodul immer laden*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/11 | Binärwert | 0 oder 1 | Steuerung | Nur Lesen |

Diese Funktion ist nur verfügbar, wenn dieses Bit im Steuerprogramm gesetzt wird (1), bevor das Programm auf das Speichermodul herun- tergeladen wird. Wenn dieses Bit im Speichermodul gesetzt ist und die Spannungsversorgung eingeschaltet wird, lädt die Steuerung das Speichermodulprogramm herunter.

Der Steuerungsmodus nach der Übertragung wird durch den Steue- rungsmodusschalter *(nur MicroLogix 1500)* und das Auswahl-Bit für das Einschaltmodusverhalten (S:1/12) bestimmt.

Siehe auch: „LA – Immer laden“ auf Seite 3-9.

*Einschaltmodusverhalten*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/12 | Binärwert | 0 oder 1 | Steuerung | Nur Lesen |

Wenn das Bit für das Einschaltmodusverhalten gelöscht ist (0 = letzter Status), hängt der Einschaltmodus von folgenden Faktoren ab:

- Position des Modusschalters *(nur MicroLogix 1500)*
- Status des Bits für Halt wegen schwerem Fehler (S:1/13)
- Modus beim vorherigen Ausschaltvorgang

Wenn das Bit für das Einschaltmodusverhalten gesetzt ist (1 = Ausführung), hängt der Einschaltmodus von folgenden Faktoren ab:

- Position des Modusschalters *(nur MicroLogix 1500)*
- Status des Bits für Halt wegen schwerem Fehler (S:1/13)

**WICHTIG** Wenn beim Einschalten unabhängig von vorherigen Fehlerbedingungen der Run-Modus aktiviert werden soll, muss das Bit S:1/8 (Fehler überbrücken) gesetzt werden, damit das Bit für Halt wegen schwerem Fehler vor der Bestimmung der Einschaltmodus rückgesetzt wird.

Die folgende Tabelle zeigt den Einschaltmodus unter verschiedenen Bedingungen

| **MicroLogix 1200** | **Halt wegen schwerem Fehler** | **Einschaltmodus verhalten** | **Modus beim letzten Ausschaltvorgang** | **Einschaltmodus** |
|---|---|---|---|---|
| **dezentral** | unwahr | letzter Status | REM Herunterladen, Herunterladen, REM Programm, Programm oder Testmodus | REM Programm-Modus |
| | | | REM-Suspend-Zustand oder Suspend-Zustand | REM-Suspend-Zustand |
| | | | REM Run oder Run | REM Run |
| | | Ausführung | ignorieren | REM Run |
| | wahr | ignorieren | ignorieren | REM Programm mit Fehler |
| **MicroLogix 1500 - Position des Modus-schalters beim Einschalten** | **Halt wegen schwerem Fehler** | **Einschaltmodus verhalten** | **Modus beim letzten Ausschaltvorgang** | **Einschaltmodus** |
| **Programm** | unwahr | ignorieren | ignorieren | Programm |
| | wahr | | | Programm mit Fehler |
| **dezentral** | unwahr | letzter Status | REM Herunterladen, Herunterladen, REM Programm, Programm oder Testmodus | REM Programm-Modus |
| | | | REM-Suspend-Zustand oder Suspend-Zustand | REM-Suspend-Zustand |
| | | | REM Run oder Run | REM Run |
| | | Ausführung | ignorieren | REM Run |
| | wahr | ignorieren | ignorieren | REM Programm mit Fehler |
| **Ausführung** | unwahr | letzter Status | REM-Suspend-Zustand oder Suspend-Zustand | Suspend-Zustand |
| | | | Jeder Modus außer REM Suspend oder Suspend | Ausführung |
| | | Ausführung | ignorieren | Ausführung |
| | wahr | ignorieren | ignorieren | Run mit Fehler(1) |

(1) Run mit Fehler ist eine Fehlerbedingung, die dem Steuerungsmodus Programm mit Fehler entspricht (Ausgänge werden rückgesetzt und das Steuerungsprogramm wird nicht ausgeführt). Sobald das Flag für Halt wegen schwerem Fehler gelöscht wird, wird der Run-Modus aktiviert.

Siehe auch: „MB – Modusverhalten“ auf Seite 3-9.

### *Halt wegen schwerem Fehler*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/13 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird von der Steuerung gesetzt (1), wenn ein schwerwie- gender Fehler festgestellt wurde. Die Steuerung wird in einen Fehlerzustand versetzt, und das Wort S:6 enthält den Fehlercode, der zur Diagnose der Bedingung verwendet werden kann. Wenn das Bit S:1/13 gesetzt ist, reagiert der Prozessor folgendermaßen:

- Alle Ausgänge werden ausgeschaltet und die LED FAULT blinkt,
- oder, die Anwenderfehlerroutine wird ausgeführt, damit das Steuerprogramm die Fehlerbedingung beseitigen kann. Wenn das Bit S:1/13 von der Anwenderfehlerroutine gelöscht und die Fehler- bedingung beseitigt wird, fährt die Steuerung mit der Ausführung des Steuerprogramms fort. Wenn der Fehler nicht gelöscht werden kann, werden die Ausgänge gelöscht, die Steuerung beendet den Ausführungsmodus, und die LED FAULT blinkt.

**ACHTUNG**

Wenn Sie das Bit für Halt wegen schwerem Fehler (S:1/13) löschen, während sich der Steuerungsmodus- schalter *(nur MicroLogix 1500)* in der Stellung RUN befindet, wird sofort der RUN-Modus aktiviert.

### *Zukünftiger Zugriff (OEM-Sperre)*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/14 | Binärwert | 0 oder 1 | Status | Nur Lesen |

Wenn dieses Bit gesetzt ist (1), muss das Programmiergerät über eine exakte Kopie des Steuerungsprogramms verfügen.

Weitere Informationen hierzu finden Sie unter „Zukünftigen Zugriff zulassen (OEM-Sperre)“ auf Seite 2-13.

### *Erste Abfrage*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:1/15 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Wenn dieses Bit vom Prozessor gesetzt wurde (1), bedeutet dies, dass das Anwenderprogramm zu diesem Zeitpunkt das erste Mal abgefragt wird (dies geschieht nach Aktivierung des Ausführungsmodus). Die Steuerung setzt dieses Bit nach der ersten Abfrage zurück.

**HINWEIS** Das Erstabfrage-Bit (S:1/15) wird während der Ausfüh- rung der Einschaltschutz-Fehlerroutine gesetzt. Weitere Informationen finden Sie unter S:1/9.

## STI-Modus

*STI anstehend*

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:2/0 | Binärwert | 0 oder 1 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird an STI:0/UIP dupliziert. Weitere Informationen hierzu finden Sie unter „STI-Funktionsfile verwenden“ auf Seite 18-12.

*STI aktiviert*

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:2/1 | Binärwert | 0 oder 1 | Steuerung | Lesen/Schreiben |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird an STI:0/TIE dupliziert. Weitere Informationen hierzu finden Sie unter „STI-Funktionsfile verwenden“ auf Seite 18-12.

*STI-Ausführung*

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:2/2 | Binärwert | 0 oder 1 | Steuerung | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird an STI:0/UIX dupliziert. Weitere Informationen hierzu finden Sie unter „STI-Funktionsfile verwenden“ auf Seite 18-12.

## Speichermodul-Programmvergleich

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:2/9 | Binärwert | 0 oder 1 | Steuerung | Nur Lesen |

Wenn dieses Bit in der Steuerung gesetzt ist (1), kann der Ausfüh- rungsmodus der Steuerung nur aktiviert werden, wenn das Anwen- derprogramm in der Steuerung mit dem des Speichermoduls identisch ist.

Wenn die Anwenderprogramme in der Steuerung und dem Spei- chermodul nicht identisch sind oder das Speichermodul nicht vorhanden ist, tritt bei jedem Versuch der Aktivierung des Ausfüh- rungsmodus ein Steuerungsfehler mit Fehlercode 0017H auf.

RTC-Module unterstützen den Programmvergleich nicht. Wenn der Programmvergleich aktiviert ist und ein reines RTC-Modul installiert wurde, kann der Ausführungsmodus der Steuerung nicht aktiviert werden.

Siehe auch: „LPC – Programmvergleich“ auf Seite 3-8.

## Auswahl der mathematischen Überlauffunktion

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:2/14 | Binärwert | 0 oder 1 | Steuerung | Lesen/Schreiben |

Setzen Sie dieses Bit (1), wenn eine 32-Bit-Addition und -Subtraktion verwendet werden soll. Wenn S:2/14 gesetzt ist und das Ergebnis eines ADD-, SUB-, MUL- oder DIV-Befehls nicht in der Zieladresse dargestellt werden kann (Unterschreitung oder Überlauf), tritt Folgendes ein:

- das Überlauf-Bit S:0/1 wird gesetzt,
- das Überlauferkennungs-Bit S:5/0 wird gesetzt und
- die Zieladresse enthält die abgeschnittenen niederwertigsten 16 oder 32 Bits des Ergebnisses ohne Vorzeichen.

Im Standardzustand ist das Bit S:2/14 gelöscht (0). Wenn S:2/14 gelöscht ist (0) und das Ergebnis eines ADD-, SUB-, MUL- oder DIV-Befehls nicht in der Zieladresse dargestellt werden kann (Unterschreitung oder Überlauf), tritt Folgendes ein:

- das Überlauf-Bit S:0/1 wird gesetzt,
- das Überlauferkennungs-Bit S:5/0 wird gesetzt und
- die Zieladresse enthält +32 767 (Wort) oder +2 147 483 647 (Doppelwort), wenn das Ergebnis positiv ist; oder –32 768 (Wort) bzw. – 2 147 483 648 (Doppelwort), wenn das Ergebnis negativ ist.

Zum Schutz vor einer versehentlichen Änderung dieser Einstellung programmieren Sie einen unbedingten OTL-Befehl auf S:2/14, um eine neue mathematische Überlauffunktion zu aktivieren. Programmieren Sie einen unbedingten OTU-Befehl auf S:2/14, um den ursprünglichen Mathematiküberlauf zu aktivieren.

## Watchdog-Abfragezeit

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:3H | Byte | 2 bis 255 | Steuerung | Lesen/Schreiben |

In diesem Byte ist der Wert gespeichert, der festlegt, wie viele 10-ms-Intervalle während eines Programmabfragezyklus zulässig sind. Die Zeitgenauigkeit liegt zwischen -10 ms und +0 ms. Damit resultiert beispielsweise der Wert 2 in einem Zeitablauf zwischen 10 und 20 ms.

Stimmt der Wert der Programmabfrage mit dem Watchdog-Wert überein, wird ein schwerwiegender Watchdog-Fehler (Code 0022H) ausgelöst.

## Freilaufender Takt

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:4 | Binärwert | 0 bis FFFF | Status | Lesen/Schreiben |

Dieses Register enthält einen freilaufenden Zähler, der in Intervallen von 100 µs zählt. Dieses Wort wird bei Aktivierung eines Ausführungs- modus gelöscht (0).

## Bits für geringfügige Fehler

*Überlauferkennungsbit*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/0 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Ist dieses Bit bei der Ausführung eines END- oder TND- Befehls gesetzt (1), wird ein schwerwiegender Fehler (0020) generiert. Um das Auftreten dieser schwerwiegender Fehler zu vermeiden, werten Sie den Zustand dieses Bits im Anschluss an einen Mathematikbefehl (ADD, SUB, MUL, DIV, NEG, SCL, TOD oder FRD) aus, ergreifen Sie entsprechende Maßnahmen, und löschen Sie anschließend das Bit S:5/0 mit einem OTU-Befehl mit S:5/0.

*Steuerregisterfehler*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/2 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Die Befehle LFU, LFL, FFU, FFL, BSL, BSR, SQO, SQC und SQL können diesen Fehler generieren. Wenn dieses Bit S:5/2 gesetzt ist (1), wurde das Fehler-Bit eines Steuerworts, das von dem Befehl ver- wendet wird, gesetzt.

Ist dieses Bit bei der Ausführung eines END- oder TND- Befehls gesetzt, wird ein schwerwiegender Fehler (0020) generiert. Um das Auftreten dieser schwerwiegenden Fehler zu vermeiden, werten Sie nach einem Steuerregisterbefehl den Zustand dieses Bits aus, ergreifen Sie entsprechende Maßnahmen, und löschen Sie anschließend das Bit S:5/2 mit einem OTU-Befehl (S:5/2).

*Schwerwiegender Fehler in Benutzerfehlerroutine*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/3 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Wenn dieses Bit gesetzt ist (1), bezeichnet der Code für schwerwie- gende Fehler (S:6) den schwerwiegenden Fehler, der bei der Verarbeitung der Anwenderfehlerroutine aufgrund eines anderen schwerwiegenden Fehlers aufgetreten ist.

*Speichermodul laden*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/8 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Wenn dieses Bit durch die Steuerung gesetzt wird (1), wurde ein Speichermodulprogramm übertragen, weil S:1/10 (Speichermodul laden bei Fehler oder Standardprogramm) oder S:1/11 (Speicher- programm immer laden) in einem angeschlossenen Speicher- modul-Anwenderprogramm gesetzt ist. Dieses Bit wird nicht von der Steuerung rückgesetzt (0).

Das Programm kann den Status dieses Bits bei der ersten Abfrage (mit Bit S:1/15) nach Aktivierung des Ausführungsmodus prüfen, um fest- zustellen, ob das Speichermodul-Anwenderprogramm nach dem Einschalten übertragen wurde. Diese Information ist vor allem hilf- reich bei Anwendungen, die speichernde Daten enthalten und im Speichermodul das Bit S:1/10 oder S:1/11 gesetzt ist.

*Fehlende Übereinstimmung bei Speichermodulkennwörtern*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/9 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt (1), wenn beim Einschalten das Bit „Immer laden“ gesetzt ist und das Kennwort der Steuerung und des Speicher- moduls nicht übereinstimmen.

Weitere Informationen hierzu finden Sie unter „Kennwortschutz“ auf Seite 2-11.

*STI-Verlust*

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/10 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird an STI:0/UIL dupliziert. Weitere Informationen hierzu finden Sie unter „STI-Funktionsfile verwenden“ auf Seite 18-12.

*Remanente Daten verloren (nur MicroLogix 1200)*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/11 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt (1), sobald remanente Daten verloren gehen. Dieses Bit bleibt gesetzt, bis es von dem Anwender gelöscht (0) wird. Die Steuerung prüft die remanenten Daten beim Einschalten. Wenn dabei ungültige Anwenderdaten festgestellt werden, wird das Bit für verloren gegangene remanente Daten gesetzt. Die Daten in der Steue- rung sind die Werte, die sich bei der letzten Übertragung des Programms auf die Steuerung in dem Programm befanden. Wenn das Bit für verloren gegangene remanente Daten gesetzt ist, tritt bei Akti- vierung eines Ausführungsmodus ein Fehler auf, sofern das Bit für die Fehlerüberschreibung (S:1/8) nicht gesetzt ist.

*Ladestatus Batterie (nur MicroLogix 1500)*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/11 | Binärwert | 0 oder 1 | Status | Nur Lesen |

Dieses Bit wird bei niedrigem Batterieladezustand gesetzt (1).

**WICHTIG** Setzen Sie sofort eine Ersatzbatterie ein. Weitere Informationen finden Sie in Ihrem Hardwarehandbuch.

Siehe auch: „RTC-Batteriebetrieb“ auf Seite 3-4.

*Eingangsfilterwahl geändert*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/13 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Dieses Bit wird gesetzt (1), wenn die Auswahl für den diskreten Eingangsfilter im Steuerprogramm nicht mit der Hardware kompatibel ist.

*Fehler-Bit für die ASCII-Zeichenkettenbearbeitung*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:5/15 | Binärwert | 0 oder 1 | Status | Lesen |

Dieses Bit wird gesetzt (1), wenn eine ungültige Zeichenkettenlänge auftritt. Ist S:5/15 gesetzt, wird der Fehler bezüglich der ungültigen Zeichenkettenlänge (1F39H) in das Haupt-Fehlercodewort (S:6) geschrieben.

Dieses Bit gilt für die MicroLogix 1200- und 1500-Steuerungen der Serie B.

## Fehlercode für schwerwiegende Fehler

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:6 | Wort | 0 bis FFFF | Status | Lesen/Schreiben |

Dieses Register zeigt einen Wert an, mit dessen Hilfe die Ursache eines Fehlers ermittelt werden kann. Weitere Hinweise zur Fehlersuche und -beseitigung finden Sie unter „Erkennen von Steuerungsfehlern“ auf Seite D-1.

## Suspend-Code

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:7 | Wort | -32768 bis +32767 | Status | Lesen/Schreiben |

Bei der Ausführung eines SUS-Befehls (Suspend) durch die Steuerung wird der SUS-Code in diese Adresse, S:7, geschrieben. Damit werden die Bedingungen in der Anwendung festgehalten, die diesen Warte- zustand verursacht haben. Dieser Wert wird von der Steuerung nicht gelöscht.

Verwenden Sie den SUS-Befehl zur Fehlersuche bei der Inbetrieb- nahme oder als Laufzeitdiagnose zur Erkennung von Systemfehlern.

## Suspend-File

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:8 | Wort | 0 bis 255 | Status | Lesen/Schreiben |

Bei der Ausführung eines SUS-Befehls (Wartezustand) durch die Steuerung wird der SUS-File in diese Adresse, S:8, geschrieben. Damit werden die Bedingungen in der Anwendung festgehalten, die diesen Wartezustand verursacht haben. Dieser Wert wird von der Steuerung nicht gelöscht.

Verwenden Sie den SUS-Befehl zur Fehlersuche bei der Inbetrieb- nahme oder als Laufzeitdiagnose zur Erkennung von Systemfehlern.

## Aktive Netznoten (Netzknoten 0 bis 15)

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:9 | Wort | 0 bis FFFF | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird im Kommunikations-Statusfile dupliziert (CSx:0,27). Weitere Informationen hierzu finden Sie unter „Block aktive Netzknotentabelle“ auf Seite 3-18.

## Aktive Netzknoten (Netzknoten 16 bis 31)

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:10 | Wort | 0 bis FFFF | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird im Kommunikations-Statusfile dupliziert (CSx:0,28). Weitere Informationen hierzu finden Sie unter „Block aktive Netzknotentabelle“ auf Seite 3-18.

## Rechenregister

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:13 (tiefes Byte) | Wort | -32768 bis +32767 | Status | Lesen/Schreiben |
| S:14 (hohes Byte) | Wort | -32768 bis +32767 | Status | Lesen/Schreiben |

Diese beiden Worte werden in Verbindung mit den Mathematik- befehlen MUL, DIV, FRD und TOD verwendet. Der Rechenregister- wert wird bei Ausführung des Befehls geprüft und bleibt bis zur Ausführung des nächsten MUL-, DIV-, FRD- oder TOD-Befehls im Anwenderprogramm gültig.

## Netzknotenadresse

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:15 (niedriges Byte) | Byte | 0 bis 255 | Status | Nur Lesen |

(1) Der Zugriff auf dieses Byte ist nur über die Kontaktplanlogik möglich. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird in dem Kommunikations-Statusfile dupliziert (CSx:0,5/0 bis CSx:0,5/7). Weitere Informationen hierzu finden Sie unter „Block zum allgemeinen Kanalstatus“ auf Seite 3-15.

## Baudrate

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:15 (hohes Byte) | Byte | 0 bis 255 | Status | Nur Lesen |

(1) Der Zugriff auf dieses Byte ist nur über die Kontaktplanlogik möglich. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird in dem Kommunikations-Statusfile dupliziert (CSx:0,5/8 bis CSx:0,5/15). Weitere Informationen hierzu finden Sie unter „Block zum allgemeinen Kanalstatus“ auf Seite 3-15.

## Maximale Scanzeit

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:22 | Wort | 0 bis 32767 | Status | Lesen/Schreiben |

Dieses Wort enthält das größte ermittelte Zeitintervall zwischen zwei aufeinanderfolgenden Programmabfragen.

Dabei vergleicht die Steuerung jeden Abfragewert mit dem in S:22 gespeicherten Wert. Wenn ein Abfragewert größer als der vorherige ist, wird der höhere Wert in S:22 gespeichert.

Dieser Wert zeigt die Programmabfragezeit des längsten Abfragezyklus (in Schritten von 100 µs) an. Die Auflösung beträgt -100 µs bis +0 µs. Beispielsweise bedeutet der Wert 9, dass der längste Programmab- fragezyklus eine Länge von 800 bis 900 µs hatte.

## Filenummer für Benutzerfehler-Routine

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:29 | Wort | 0 bis 255 | Status | Nur Lesen |

Mit diesem Register wird festgelegt, welches Unterprogramm bei einem Anwenderfehler ausgeführt wird.

## STI-Sollwert

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:30 | Wort | 0 bis 65535 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird an STI:0/SPM dupliziert. Weitere Informationen hierzu finden Sie unter „STI-Funktionsfile verwenden“ auf Seite 18-12.

## STI-Filenummer

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:31 | Wort | 0 bis 65535 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird an STI:0/PFN dupliziert. Weitere Informationen hierzu finden Sie unter „STI-Funktionsfile verwenden“ auf Seite 18-12.

## Kommunikation Kanal 0

*Eingehender Befehl im Wartezustand*

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:33/0 | Binärwert | 0 oder 1 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird im Kommunikations-Statusfile unter CS0:0.4/0 dupliziert. Weitere Informationen hierzu finden Sie unter „Block zum allgemeinen Kanalstatus“ auf Seite 3-15.

*Nachrichtenantwort anstehend*

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:33/1 | Binärwert | 0 oder 1 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird im Kommunikations-Statusfile unter CS0:0.4/1 dupliziert. Weitere Informationen hierzu finden Sie unter „Block zum allgemeinen Kanalstatus“ auf Seite 3-15.

*Ausgehender Nachrichten-Befehl anstehend*

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:33/2 | Binärwert | 0 oder 1 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird im Kommunikations-Statusfile unter CS0:0.4/2 dupliziert. Weitere Informationen hierzu finden Sie unter „Block zum allgemeinen Kanalstatus“ auf Seite 3-15.

*Kommunikationsmodusauswahl*

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:33/3 | Binärwert | 0 oder 1 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird im Kommunikations-Statusfile unter CS0:0.4/3 dupliziert. Weitere Informationen hierzu finden Sie unter „Block zum allgemeinen Kanalstatus“ auf Seite 3-15.

*Kommunikation aktiv*

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:33/4 | Binärwert | 0 oder 1 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird im Kommunikations-Statusfile unter CS0:0.4/4 dupliziert. Weitere Informationen hierzu finden Sie unter „Block zum allgemeinen Kanalstatus“ auf Seite 3-15.

*Umschaltabfrage-Bit*

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:33/9 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Der Status dieses Bits wird am Ende jeder Abfrage durch die Steue- rung geändert. Bei Aktivierung des Ausführungsmodus wird das Bit rückgesetzt.

## Letzte 100 µs Scanzeit

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:35 | Wort | 0 bis 32767 | Status | Lesen/Schreiben |

Dieses Register zeigt die Programmabfragezeit des letzten Scanzyklus (in Schritten von 100 µs) an.

## Datenfile-Überschreibschutz inaktiv

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:36/10 | Binärwert | 0 oder 1 | Status | Lesen/Schreiben |

Wenn dieses Bit gelöscht ist (0), wurden bei der letzten Programmübertragung auf die Steuerung keine geschützten Datenfiles in der Steuerung überschrieben oder das heruntergeladene Programm enthielt keine geschützten Datenfiles.

Wenn dieses Bit gesetzt ist (1), wurden die Daten überschrieben. Weitere Informationen hierzu finden Sie unter „Bedingungen für Fileschutz bei Anwenderprogrammübertragung“ auf Seite 2-9.

Weitere Informationen hierzu finden Sie unter „Festlegen der Schutzfunktion für heruntergeladene Files“ auf Seite 2-8.

## RTC-Jahr

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:37 | Wort | 1998 bis 2097 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird in dem RTC-Funktionsfile (Echtzeituhr) unter RTC:0,YR dupliziert. Weitere Informationen hierzu finden Sie unter „Echtzeituhr-Funktionsfile“ auf Seite 3-3.

## RTC-Monat

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:38 | Wort | 1 bis 12 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird in dem RTC-Funktionsfile (Echtzeituhr) unter RTC:0,MON dupliziert. Weitere Informationen hierzu finden Sie unter „Echtzeituhr-Funktionsfile“ auf Seite 3-3.

## RTC-Tag

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:39 | Wort | 1 bis 31 | Status | Nur Lesen |

(1) Dieses Bit kann nur über die Kontaktplanlogik gesetzt werden. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird in dem RTC-Funktionsfile (Echtzeituhr) unter RTC:0,DAY dupliziert. Weitere Informationen hierzu finden Sie unter „Echtzeituhr-Funktionsfile“ auf Seite 3-3.

## RTC-Stunden

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:40 | Wort | 0 bis 23 | Status | Nur Lesen |

(1) Der Zugriff auf dieses Wort ist nur über die Kontaktplanlogik möglich. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird in dem RTC-Funktionsfile (Echtzeituhr) unter RTC:0,HR dupliziert. Weitere Informationen hierzu finden Sie unter „Echtzeituhr-Funktionsfile“ auf Seite 3-3.

## RTC-Minuten

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:41 | Wort | 0 bis 59 | Status | Nur Lesen |

(1) Der Zugriff auf dieses Wort ist nur über die Kontaktplanlogik möglich. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird in dem RTC-Funktionsfile (Echtzeituhr) unter RTC:0,MIN dupliziert. Weitere Informationen hierzu finden Sie unter „Echtzeituhr-Funktionsfile“ auf Seite 3-3.

## RTC-Sekunden

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:42 | Wort | 0 bis 59 | Status | Nur Lesen |

(1) Der Zugriff auf dieses Wort ist nur über die Kontaktplanlogik möglich. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird in dem RTC-Funktionsfile (Echtzeituhr) unter RTC:0,SEC dupliziert. Weitere Informationen hierzu finden Sie unter „Echtzeituhr-Funktionsfile“ auf Seite 3-3.

## RTC-Wochentag

| Adresse(1) | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:53 | Wort | 0 bis 6 | Status | Nur Lesen |

(1) Der Zugriff auf dieses Wort ist nur über die Kontaktplanlogik möglich. Ein Zugriff über Kommunikationsverbindungen (z. B. einen Nachrichtenbefehl von einem anderen Gerät) ist nicht möglich.

Diese Adresse wird in dem RTC-Funktionsfile (Echtzeituhr) unter RTC:0,DOW dupliziert. Weitere Informationen hierzu finden Sie unter „Echtzeituhr-Funktionsfile“ auf Seite 3-3.

## Bestellnummer Betriebssystem

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:57 | Wort | 0 bis 32767 | Status | Nur Lesen |

Dieses Register bezeichnet die Bestellnummer des in der Steuerung verwendeten Betriebssystems.

## Betriebssytemserie

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:58 | ASCII | A bis Z | Status | Nur Lesen |

Dieses Register enthält die Serienbezeichnung (Buchstabe) des in der Steuerung verwendeten Betriebssystems.

## Betriebssytem-FRN

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:59 | Wort | 0 bis 32767 | Status | Nur Lesen |

Dieses Register enthält die FRN des in der Steuerung verwendeten Betriebssystems.

## Prozessor-Bestellnummer

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:60 | ASCII | „A“ bis „ZZ“ | Status | Nur Lesen |

Dieses Register bezeichnet die Bestellnummer des Prozessors.

## Prozessorserie

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:61 | ASCII | A bis Z | Status | Nur Lesen |

Dieses Register bezeichnet die Serie des Prozessors.

## Prozessorversion

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:62 | Wort | 0 bis 32767 | Status | Nur Lesen |

Dieses Register enthält die Revisionsnummer (Start-FRN) des Prozessors.

## Benutzerprogramm-Funktionstyp

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:63 | Wort | 0 bis 32767 | Status | Nur Lesen |

Dieses Register bezeichnet die Funktionalität des in der Steuerung verwendeten Anwenderprogramms.

## Compiler-Revision – Build-Nummer

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:64 (niedriges Byte) | Byte | 0 bis 255 | Status | Nur Lesen |

Dieses Register enthält die Build-Nummer des Compilers, mit dem das Programm in der Steuerung erstellt wurde.

## Compiler-Revision – Versionsnummer

| Adresse | Datenformat | Bereich | Typ | Anwenderprogrammzugriff |
|---|---|---|---|---|
| S:64 (hohes Byte) | Byte | 0 bis 255 | Status | Nur Lesen |

Dieses Register enthält die Versionsnummer des Compilers, mit dem das Programm in der Steuerung erstellt wurde.

# Fehlermeldungen und Fehlercodes

In diesem Kapitel wird beschrieben, wie Sie Störungen in der Steuerung finden und beseitigen. Dabei werden folgende Themen erläutert:

- Steuerungsfehler erkennen
- Anfordern der Unterstützung von Rockwell Automation

## Erkennen von Steuerungsfehlern

Während der Ausführung eines Programms können innerhalb des Betriebssystems oder des Programms Fehler auftreten. Wenn dies der Fall ist, können der Fehlerzustand und die Vorgehensweise für dessen Beseitigung auf mehrere Weisen ermittelt werden. In diesem Abschnitt werden Möglichkeiten zur Fehlerbeseitigung beschrieben. Außerdem werden die möglichen Fehleranzeigen und die empfohlenen Abhilfe- maßnahmen in einer Liste dargestellt.

### Automatische Fehlerbeseitigung

Wenn das Bit S:1/8 zur Fehlerüberbrückung beim Einschalten im Statusfile gesetzt ist, können Sie Fehler automatisch durch Aus- und Wiedereinschalten der Steuerung beseitigen.

Außerdem können Sie die Steuerung so konfigurieren, dass Fehler beseitigt und der RUN-Modus aktiviert wird, sobald die Steuerung aus- und wieder eingeschaltet wird. Diese Funktion können Originalteile- hersteller in ihre jeweiligen Geräte integrieren, damit Endbenutzer die Steuerung zurücksetzen können. Bei einem Steuerungsfehler kann die Steuerung durch Aus- und Wiedereinschalten der Maschine zurückge- setzt werden. Zu diesem Zweck müssen Sie im Statusfile folgende Bits setzen:

- S2:1/8 - Fehler beim Einschalten beseitigen
- S2:1/12 - Modusverhalten

Wenn die Fehlerbedingung nach dem Aus- und Einschalten der Spannungsversorgung weiterhin besteht, wird erneut der Fehlermodus aktiviert. Weitere Informationen zu Status-Bits finden Sie unter „Systemstatusfile“ auf Seite C-1.

**HINWEIS** Sie können anwendungsspezifische, schwerwiegende Fehler definieren; geben Sie hierfür einen eindeutigen Wert in S:6 ein, und setzen Sie dann das Bit S:1/13, um eine Verwendung der Systemcodes zu verhindern. Die empfohlenen Werte für anwenderdefinierte Fehler liegen in einem Bereich von FF00 bis FF0F.

## Manuelle Fehlerbeseitigung mit der Fehlerroutine

Wenn ein behebbarer oder nicht behebbarer Anwenderfehler auftritt, wird unter Umständen die Anwenderfehlerroutine ausgeführt. Bei behebbaren Fehlern kann das Problem mit Hilfe des Unterprogramms behoben und das Fehler-Bit S:1/13 gelöscht werden. Der Steuerungs- betrieb wird dann im Run- oder im Test-Modus fortgesetzt.

Bei Fehlern, die nicht vom Anwender verursacht wurden, wird das Unterprogramm nicht ausgeführt. Informationen zur Erstellung einer Anwenderfehlerroutine finden Sie unter „Anwenderfehlerroutine“ auf Seite 18-6.

## Fehlermeldungen

Dieser Abschnitt enthält die Fehlermeldungen, die während des Betriebs der speicherprogrammierbaren Steuerungen MicroLogix 1200 und MicroLogix 1500 auftreten können. Jeder Tabelleneintrag enthält den Fehlercode, eine Beschreibung des Fehlers, die wahrscheinliche Ursache und die empfohlenen Abhilfemaßnahmen.

| Fehlercode (Hex) | Meldung | Beschreibung | Fehlerklassifizierung | Empfohlene Abhilfemaßnahme |
|---|---|---|---|---|
| 0001 | NVRAM ERROR | Das Standardprogramm wurde in den Speicher der Steuerung geladen. Dies erfolgt in folgenden Situationen:<br>• bei einem Stromausfall während des Herunterladens oder der Übertragung eines Programms aus dem Speichermodul.<br>• Fehler bei RAM-Test<br>• Fehler bei FLASH-Test *(nur MicroLogix 1200).* | Nicht anwenderbezogen | • Laden Sie das Programm erneut herunter, oder übertragen Sie es erneut.<br>• Batterieanschluss prüfen *(nur MicroLogix 1500).*<br>• Wenn der Fehler weiterhin auftritt, wenden Sie sich an die Rockwell Automation-Niederlassung vor Ort. |
| 0002 | UNEXPECTED RESET | • Die Steuerung wurde aufgrund von Störspannungen oder eines internen Hardwarefehlers unerwartet zurückgesetzt.<br>• Das Standardprogramm wird geladen. *(nur MicroLogix 1500)*<br>• Remanente Daten gingen verloren. Siehe Seite C-13. *(nur MicroLogix 1200)* | Nicht anwenderbezogen | • Prüfen Sie die in dem Benutzerhandbuch der Steuerung beschriebenen Empfehlungen zur Erdung und zum Überspannungsschutz.<br>• Batterieanschluss prüfen *(nur MicroLogix 1500).*<br>• Wenn der Fehler weiterhin auftritt, wenden Sie sich an die Rockwell Automation-Niederlassung vor Ort. |
| 0003 | MEMORY MODULE USER PROGRAM IS CORRUPT | Speicherfehler des Speichermoduls. Dieser Fehler kann auch beim Übergang in den Run-Modus auftreten. | Nicht anwenderbezogen | Programmieren Sie das Speichermodul erneut. Wenn der Fehler erneut auftritt, tauschen Sie das Speichermodul aus. |
| 0004 | MEMORY INTEGRITY ERROR | Beim Einschalten der Steuerung wurde der ROM- oder der RAM-Speicher beschädigt. | Nicht anwenderbezogen | • Schalten Sie die Stromzufuhr zur Steuerung aus und wieder ein. Laden Sie das Programm dann erneut herunter, und starten Sie das System erneut.<br>• Prüfen Sie die in dem Benutzerhandbuch der Steuerung beschriebenen Empfehlungen zur Erdung und zum Überspannungsschutz.<br>• Wenn der Fehler weiterhin auftritt, wenden Sie sich an die Rockwell Automation-Niederlassung vor Ort. |
| 0005 | RETENTIVE DATA IS LOST *(nur MicroLogix 1200)* | Remanente Daten gingen verloren. Siehe Seite C-13. | Behebbar | Wenn der Fehler weiterhin auftritt, wenden Sie sich an die Rockwell Automation-Niederlassung vor Ort. |
| 0006 | MEMORY MODULE HARDWARE FAULT | Hardwarefehler im Speichermodul oder Inkompatibilität zwischen Speichermodul und Betriebssystem. | Nicht anwenderbezogen | • Installieren Sie die aktuelle Version des Betriebssystems, damit dieses mit dem Speichermodul kompatibel ist.<br>• Verwenden Sie ein neues Speichermodul. |
| 0007 | MEMORY MODULE TRANSFER ERROR | Fehler bei der Speichermodulübertragung | Nicht anwenderbezogen | Führen Sie die Übertragung erneut aus. Wenn der Fehler erneut auftritt, tauschen Sie das Speichermodul aus. |

| Fehlercode (Hex) | Meldung | Beschreibung | Fehlerklassifizierung | Empfohlene Abhilfemaßnahme |
|---|---|---|---|---|
| 0008 | FATAL INTERNAL SOFTWARE ERROR | Ein unerwarteter Software-Fehler ist aufgetreten. | Nicht anwenderbezogen | • Schalten Sie die Stromzufuhr zur Steuerung aus und wieder ein. Laden Sie das Programm dann erneut herunter, und initialisieren Sie die erforderlichen Daten neu.<br>• Starten Sie das System.<br>• Prüfen Sie die in dem Benutzerhandbuch der Steuerung beschriebenen Empfehlungen zur Erdung und zum Überspannungsschutz.<br>• Wenn der Fehler weiterhin auftritt, wenden Sie sich an die Rockwell Automation-Niederlassung vor Ort. |
| 0009 | FATAL INTERNAL HARDWARE ERROR | Ein unerwarteter Hardware-Fehler ist aufgetreten. | Nicht anwenderbezogen | • Schalten Sie die Stromzufuhr zur Steuerung aus und wieder ein. Laden Sie das Programm dann erneut herunter, und initialisieren Sie die erforderlichen Daten neu.<br>• Starten Sie das System.<br>• Prüfen Sie die in dem Benutzerhandbuch der Steuerung beschriebenen Empfehlungen zur Erdung und zum Überspannungsschutz.<br>• Wenn der Fehler weiterhin auftritt, wenden Sie sich an die Rockwell Automation-Niederlassung vor Ort. |
| 000A | OS MISSING OR CORRUPT | Das für das Anwenderprogramm erforderliche Betriebssystem ist beschädigt oder nicht vorhanden. | Nicht anwenderbezogen | • Laden Sie ein neues Betriebssystem mit ControlFlash herunter.<br>• Weitere Informationen zu den verfügbaren Betriebssystemen für Ihre Steuerung erhalten Sie bei Ihrem lokalen Rockwell Automation-Vertriebsbeauftragten. |
| 000B | BASE HARDWARE FAULT | Hardwarefehler in der Basiseinheit oder Inkompatibilität zwischen Basiseinheit und Betriebssystem. | Nicht anwenderbezogen | • Betriebssystem mit ControlFlash aufrüsten.<br>• Steuerung ersetzen *(nur MicroLogix 1200)*.<br>• Basiseinheit ersetzen *(nur MicroLogix 1500)*.<br>• Weitere Informationen zu den verfügbaren Betriebssystemen für Ihre Steuerung erhalten Sie bei Ihrem lokalen Rockwell Automation-Vertriebsbeauftragten. |
| 0011 | EXECUTABLE FILE 2 IS MISSING | Der Kontaktplanfile 2 ist nicht in dem Programm verfügbar. | Nicht anwenderbezogen | • Kompilieren und laden Sie das Anwenderprogramm erneut. |
| 0012 | LADDER PROGRAM ERROR | Die Speichersicherheit im Kontaktplan ist gestört. | Nicht anwenderbezogen | • Laden Sie das Programm erneut, oder kompilieren und laden Sie das Programm erneut. Wenn der Fehler erneut auftritt, sollten Sie sicherstellen, dass das Programm mit der RSI-Programmiersoftware entwickelt und geladen wird.<br>• Prüfen Sie die in dem Benutzerhandbuch der Steuerung beschriebenen Empfehlungen zur Erdung und zum Überspannungsschutz. |

| Fehlercode (Hex) | Meldung | Beschreibung | Fehlerklassifizierung | Empfohlene Abhilfemaßnahme |
|---|---|---|---|---|
| 0015 | I/O CONFIGURATION FILE ERROR | Die E/A-Konfiguration des Anwenderprogramms ist ungültig. | Nicht anwenderbezogen | Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. Wenn der Fehler erneut auftritt, sollten Sie sicherstellen, dass das Programm mit der RSI-Programmiersoftware entwickelt und geladen wird. |
| 0016 | STARTUP PROTECTION FAULT | Die Anwenderfehlerroutine wurde beim Einschalten, vor dem Hauptkontaktplan, ausgeführt. Bit S:1/13 (Halt wegen schwerem Fehler) wurde am Ende der Anwenderfehlerroutine nicht zurückgesetzt. Die Anwenderfehlerroutine wurde ausgeführt, weil das Bit S:1/9 beim Einschalten gesetzt war. | Behebbar | • Setzen Sie entweder Bit S:1/9 zurück, wenn sich dies mit den Anforderungen Ihrer Anwendung vereinbaren lässt, und schalten Sie in den RUN-Modus zurück, oder<br>• Setzen Sie Bit S:1/13 (Halt wegen schwerem Fehler) am Ende der Anwenderfehlerroutine zurück. |
| 0017 | NVRAM/MODULE MEMORY USER PROGRAM MISMATCH | Bit S:2/9 ist in der Steuerung gesetzt, und das Anwenderprogramm im Speichermodul stimmt nicht mit dem Anwenderprogramm in der Steuerung überein. | Nicht behebbar | Übertragen Sie das Programm aus dem Speichermodul auf die Steuerung, und schalten Sie dann in den RUN-Modus um. |
| 0018 | MEMORY MODULE USER PROGRAM INCOMPATIBLE WITH OS | Das Anwenderprogramm im Speichermodul ist nicht mit dem Betriebssystem kompatibel. | Nicht anwenderbezogen | • Installieren Sie mit ControlFlash eine neue Betriebssystemversion, die mit dem Speichermodul kompatibel ist.<br>• Verwenden Sie ein neues Speichermodul.<br>• Weitere Informationen zu den verfügbaren Betriebssystemen für Ihre Steuerung erhalten Sie bei Ihrem lokalen Rockwell Automation-Vertriebsbeauftragten. |
| 001A | USER PROGRAM INCOMPATIBLE WITH OS AT POWER-UP | Das Anwenderprogramm ist nicht mit dem Betriebssystem kompatibel. | Nicht anwenderbezogen | • Betriebssystem mit ControlFlash aufrüsten.<br>• Weitere Informationen zu den verfügbaren Betriebssystemen für Ihre Steuerung erhalten Sie bei Ihrem lokalen Rockwell Automation-Vertriebsbeauftragten. |
| 0020 | MINOR ERROR AT END-OF-SCAN DETECTED | Am Ende einer Abfrage wurde ein Bit für einen geringfügigen Fehler (Bits 0-7) in S:5 gesetzt. | Behebbar | • Korrigieren Sie die Befehlslogik, die den Fehler verursacht hat.<br>• Öffnen Sie in der Programmiersoftware das Statusfile-Fenster, und beseitigen Sie den Fehler.<br>• Aktivieren Sie den RUN-Modus. |

| Fehlercode (Hex) | Meldung | Beschreibung | Fehlerklassifizierung | Empfohlene Abhilfemaßnahme |
|---|---|---|---|---|
| 0021 | EXPANSION POWER FAIL<br>*(nur MicroLogix 1500)* | An der Erweiterungs-E/A-Bank liegt ein Fehler in der Spannungsversorgung vor.<br>Dieser Fehlercode wird generiert, wenn die Steuerung eingeschaltet ist und an der Erweiterungs-E/A-Bank keine Spannung anliegt. Dies ist ein selbstlöschender Fehlercode. Beim erneuten Einschalten der Versorgungsspannung der Erweiterungs-E/A-Bank wird der Fehler beseitigt. Siehe unten, WICHTIG. | Nicht anwenderbezogen | Legen Sie Spannung an die Erweiterungs-E/A-Bank an. Siehe unten, WICHTIG. |
| | **WICHTIG** | Wenn dieser Fehler auftritt, während sich das System im RUN-Modus befindet, wird ein Steuerungsfehler generiert. Wenn die Spannungsversorgung der Erweiterungs-E/A wieder hergestellt wird, löscht die Steuerung den Fehler und fährt im RUN-Modus fort.<br>Wenn Sie den Modusschalter umlegen, während dieser Fehler vorliegt, kann der Steuerungsbetrieb unter Umständen nach der Wiederherstellung der Spannungsversorgung der Erweiterungs-E/A nicht im RUN-Modus fortgesetzt werden.<br>Wenn eine EPF-Bedingung vorliegt und die Spannungsversorgung der Erweiterungs-E/A in Ordnung ist, legen Sie den Modusschalter auf PROGRAM und danach auf RUN um. Damit sollte der Fehler gelöscht und der Steuerungsbetrieb im RUN-Modus fortgesetzt werden. | | |
| | **HINWEIS** | Dieser Fehler kann auch auftreten, wenn in der MicroLogix 1200- oder 1500-Steuerung ein Hardware-Fehler auf dem Bus besteht. | | • Schalten Sie die Stromzufuhr zur Steuerung aus und wieder ein.<br>• Wenn der Fehler weiterhin auftritt, wenden Sie sich an die Rockwell Automation-Niederlassung vor Ort. |
| 0022 | WATCHDOG TIMER EXPIRED, SEE S:3 (SIEHE S:3) | Die Programmabfragezeit hat den den Watchdog-Zeitablauf -Wert (S:3H) überschritten. | Nicht behebbar | • Stellen Sie fest, ob das Programm in einer Schleife feststeckt, und beseitigen Sie die Störung.<br>• Erhöhen Sie den Watchdog-Zeitablauf-Wert im Statusfile. |
| 0023 | STI ERROR | Fehler in der STI-Konfiguration. | Behebbar | Stellen Sie den Fehler anhand des Fehlercodes im STI-Funktionsfile fest. |
| 0028 | INVALID OR NONEXISTENT USER FAULT ROUTINE VALUE | • Im Statusfile (S:29) wurde eine Fehlerroutinennummer eingegeben, doch entweder wurde die Fehlerroutine nicht erstellt, oder<br>• die Fehlerroutinenummer ist kleiner als 3 oder größer als 255. | Nicht anwenderbezogen | • Löschen Sie die Fehlerroutinen-Filenummer (S:29) im Statusfile, oder<br>• erstellen Sie für die im Statusfile eingetragene Filenummer (S:29) eine Fehlerroutine. Die Filenummer muss größer als 2 und kleiner als 256 sein. |
| 0029 | INSTRUCTION INDIRECTION OUTSIDE OF DATA SPACE | Indirekter Adressbezug im Kontaktplan liegt außerhalb des Gesamt-Datenfilebereichs. | Behebbar | Korrigieren Sie das Programm, um sicherzustellen, dass keine indirekten Adressbezüge außerhalb des Datenfilebereichs liegen.<br>Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. |
| 002E | EII ERROR | Fehler in der EII-Konfiguration. | Behebbar | Stellen Sie den Fehler anhand des Fehlercodes in dem EII-Funktionsfile fest. |
| 0030 | SUBROUTINE NESTING EXCEEDS LIMIT | Die Verschachtelung des JSR-Befehls übersteigt den vorhandenen Steuerungsspeicher. | Nicht anwenderbezogen | Korrigieren Sie das Anwenderprogramm, um die bestehende Verschachtelung zu reduzieren und die Einschränkungen für JSR-Befehle zu erfüllen. Laden Sie dann das Programm erneut, und aktivieren Sie den RUN-Modus. |

| Fehlercode (Hex) | Meldung | Beschreibung | Fehlerklassifizierung | Empfohlene Abhilfemaßnahme |
|---|---|---|---|---|
| 0031 | UNSUPPORTED INSTRUCTION DETECTED | Das Programm enthält einen oder mehrere Befehle, der/die nicht von der Steuerung unterstützt wird/ werden. | Nicht anwender-bezogen | • Ändern Sie das Programm, damit alle Befehle von der Steuerung unterstützt werden.<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. |
| 0032 | SQO/SQC/SQL OUTSIDE OF DATA FILE SPACE | Der Verweis eines Längen-/ Positionsparameter des Schritt-schaltwerksbefehls liegt außerhalb des Gesamt-Datenfilebereichs. | Behebbar | • Korrigieren Sie das Programm, um sicherzustellen, dass die Längen- und Positionsparameter innerhalb des Datenfilebereichs liegen.<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. |
| 0033 | BSL/BSR/FFL/FFU/ LFL/LFU CROSSED DATA FILE SPACE | Der Verweis eines Längen-/ Positionsparameter eines BSL-, BSR-, FFL-, FFU-, LFL- oder LFU-Befehls liegt außerhalb des Gesamt-Datenfilebereichs. | Behebbar | • Korrigieren Sie das Programm, um sicherzustellen, dass die Längen- und Positionsparameter innerhalb des Datenfilebereichs liegen.<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. |
| 0034 | NEGATIVE VALUE IN TIMER PRESET OR ACCUMULATOR | In den Soll oder Istwert eines Zeit-werks wurde ein negativer Wert geladen. | Behebbar | • Wenn das Programm Werte in das Ist oder Sollwertwort eines Zeitwerks überträgt, müssen Sie dafür sorgen, dass diese Werte nicht negativ sind.<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. |
| 0035 | ILLEGAL INSTRUCTION IN INTERRUPT FILE | Das Programm enthält einen TND-, REF- oder SVC-Befehl (Temporäres Ende, Auffrischung oder Kommu-nikationsbearbeitung) in einer Interrupt-Subroutine (STI, EII, HSC) oder einer Anwenderfehlerroutine. | Nicht behebbar | • Korrigieren Sie das Programm.<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. |
| 0036 | INVALID PID PARAMETER | Für einen PID-Befehlsparameter wurde ein ungültiger Wert verwendet. | Behebbar | Weitere Informationen zum PID-Befehl finden Sie unter „Prozesssteuerungsbefehl“ auf Seite 19-1. |
| 0037 | HSC ERROR | Fehler in der HSC-Konfiguration. | Behebbar | Stellen Sie den Fehlercode anhand des HSC-Funktionsfiles fest. |
| 003B | PTO ERROR | Fehler in der PTO-Konfiguration. | Behebbar oder nicht anwender-bezogen | Stellen Sie den Fehlercode anhand des PTO-Funktionsfiles fest. |
| 003C | PWM ERROR | Fehler in der PWM-Konfiguration. | Behebbar oder nicht anwender-bezogen | Stellen Sie den Fehlercode anhand des PWM-Funktionsfiles fest. |
| 003D | INVALID SEQUENCER LENGTH/POSITION | Ein Längen-/Positionsparameter eines Schrittschaltwerksbefehls (SQO, SQC, SQL) ist größer als 255. | Behebbar | Korrigieren Sie das Anwenderprogramm, kompilieren und laden Sie das Programm erneut, und schalten Sie dann in den RUN-Modus. |
| 003E | INVALID BIT SHIFT OR LIFO/FIFO PARAMETER | Ein Längenparameter eines BSR- oder BSL-Befehls ist größer als 2048 oder ein Längenparameter eines FFU-, FFL-, LFU- oder LFL-Befehls ist größer als 128 (Wortfile) oder größer als 64 (Doppelwortfile) | Behebbar | Korrigieren Sie das Anwenderprogramm, oder weisen Sie über den Speicherbele-gungsplan einen größeren Datenfilebereich zu, laden Sie dann das Programm erneut, und schalten Sie in den RUN-Modus. |

| Fehlercode (Hex) | Meldung | Beschreibung | Fehlerklassifizierung | Empfohlene Abhilfemaßnahme |
|---|---|---|---|---|
| 003F | COP/FLL OUTSIDE OF DATA FILE SPACE | Der Verweis eines Längenparameters eines COP- oder FLL-Befehls liegt außerhalb des Gesamt-Datenfilebereichs. | Behebbar | • Korrigieren Sie das Programm, um sicherzustellen, dass die Längenparameter innerhalb des Datenfilebereichs liegen.<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. |
| 0050 | CONTROLLER TYPE MISMATCH | In der Anwenderprogrammkonfiguration wurde ein bestimmter Steuerungstyp ausgewählt, der jedoch nicht mit dem tatsächlichen Steuerungstyp übereinstimmt. | Nicht anwenderbezogen | • Schließen Sie die in dem Anwenderprogramm angegebene Hardware an, oder<br>• konfigurieren Sie das Programm so, dass es mit der angeschlossenen Hardware übereinstimmt. |
| 0051 | BASE TYPE MISMATCH | In der Anwenderprogrammkonfiguration wurde ein bestimmter Hardwaretyp (AWA, BWA, BXB) ausgewählt, der jedoch nicht mit der tatsächlichen Basiseinheit übereinstimmt. | Nicht anwenderbezogen | • Schließen Sie die in dem Anwenderprogramm angegebene Hardware an, oder<br>• konfigurieren Sie das Programm so, dass es mit der angeschlossenen Hardware übereinstimmt. |
| 0052 | MINIMUM SERIES ERROR | In der Anwenderprogrammkonfiguration wurde eine mindestens erforderliche Hardwareeinheit ausgewählt, die zu einer Serie gehört, die höher ist als die Serie der tatsächlichen Hardware. | Nicht anwenderbezogen | • Schließen Sie die in dem Anwenderprogramm angegebene Hardware an, oder<br>• konfigurieren Sie das Programm so, dass es mit der angeschlossenen Hardware übereinstimmt. |
| 0070 | EXPANSION I/O TERMINATOR REMOVED<br>*(nur MicroLogix 1500)* | Der erforderliche Abschlusswiderstand der Erweiterungs-E/A wurde entfernt. | Nicht behebbar | • Überprüfen Sie den Abschlusswiderstand der Erweitungs-E/A am letzten E/A-Modul.<br>• Spannungsversorgung aus- und wieder einschalten. |
| xx71[(1)] | EXPANSION I/O HARDWARE ERROR | Die Steuerung kann nicht mit einem Erweiterungs-E/A-Modul kommunizieren. | Nicht behebbar | • Anschlüsse überprüfen.<br>• Stellen Sie fest, ob Störsignale vorhanden sind, und überprüfen Sie die Erdung.<br>• Ersetzen Sie das Modul.<br>• Spannungsversorgung aus- und wieder einschalten. |
| xx79[(1)] | EXPANSION I/O MODULE ERROR | Durch ein Erweiterungs-E/A-Modul wurde ein Fehler verursacht. | Nicht behebbar | • Überprüfen Sie den Statusfile des E/A-Moduls (IOS-File).<br>• Schlagen Sie in der Dokumentation des jeweiligen Moduls nach, um die möglichen Ursachen eines Modulfehlers zu ermitteln. |
| 0080 | EXPANSION I/O TERMINATOR REMOVED<br>*(nur MicroLogix 1500)* | Der erforderliche Abschlusswiderstand der Erweiterungs-E/A wurde entfernt. | Nicht anwenderbezogen | • Überprüfen Sie den Abschlusswiderstand der Erweitungs-E/A am letzten E/A-Modul.<br>• Spannungsversorgung aus- und wieder einschalten. |
| xx81[(1)] | EXPANSION I/O HARDWARE ERROR | Die Steuerung kann nicht mit einem Erweiterungs-E/A-Modul kommunizieren. | Nicht anwenderbezogen | • Anschlüsse überprüfen.<br>• Stellen Sie fest, ob Störsignale vorhanden sind, und überprüfen Sie die Erdung.<br>• Ersetzen Sie das Modul.<br>• Spannungsversorgung aus- und wieder einschalten. |

| Fehlercode (Hex) | Meldung | Beschreibung | Fehlerklassifizierung | Empfohlene Abhilfemaßnahme |
|---|---|---|---|---|
| 0083 | MAX I/O CABLES EXCEEDED | Die maximal zulässige Anzahl an Erweiterungs-E/A-Kabeln wurde überschritten. | Nicht anwenderbezogen | • Konfigurieren Sie das Erweiterungs-E/A-System neu, und berücksichtigen Sie dabei die maximal zulässige Kabelanzahl.<br>• Spannungsversorgung aus- und wieder einschalten. |
| 0084 | MAX I/O POWER SUPPLIES EXCEEDED | Die maximal zulässige Anzahl an Erweiterungs-E/A-Netzteilen wurde überschritten. | Nicht anwenderbezogen | • Konfigurieren Sie das Erweiterungs-E/A-System neu, und berücksichtigen Sie dabei die maximal zulässige Netzteil- anzahl. |
| 0085 | MAX I/O MODULES EXCEEDED | Die maximal zulässige Anzahl an Erweiterungs-E/A-Modulen wurde überschritten. | Nicht anwenderbezogen | • Konfigurieren Sie das Erweiterungs-E/A-System neu, und berücksichtigen Sie dabei die maximal zulässige Modulanzahl.<br>• Spannungsversorgung aus- und wieder einschalten. |
| xx86(1) | EXPANSION I/O MODULE BAUD RATE ERROR | Ein Erweiterungs-E/A-Modul kann nicht mit der in der E/A-Konfiguration des Anwenderprogramms angegebenen Baudrate kommunizieren. | Nicht anwenderbezogen | • Überprüfen Sie die Baudrate in der E/A-Konfiguration des Anwenderprogramms, und<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus, oder<br>• Ersetzen Sie das Modul.<br>• Spannungsversorgung aus- und wieder einschalten. |
| xx87(1) | I/O CONFIGURATION MISMATCH | • Die Erweiterungs-E/A-Konfiguration im Anwenderprogramm stimmt nicht mit der tatsächlichen Konfiguration überein, oder<br>• in der Erweiterungs-E/A-Konfiguration im Anwenderprogramm wird ein Modul angegeben, das nicht vorhanden ist, oder<br>• die in der Konfiguration des Erweiterungs-E/A-Moduls angegebene Datengröße übersteigt die tatsächliche Aufnahmefähigkeit des Moduls. | Nicht anwenderbezogen | • Passen Sie die E/A-Konfiguration im Anwenderprogramm an die tatsächliche Konfiguration an, oder<br>• Passen Sie bei ausgeschalteter Spannungsversorgung die tatsächliche E/A-Konfiguration an die Konfiguration im Anwenderprogramm an. |
| xx88(1) | EXPANSION I/O MODULE CONFIGURATION ERROR | Die Anzahl der Eingangs- und Ausgangsdatenworte in der Konfiguration des Anwenderprogramms ist größer als die Datenwortgröße im Erweiterungs-E/A-Modul. | Nicht anwenderbezogen | • Korrigieren Sie die E/A-Konfiguration im Anwenderprogramm, um die Anzahl der Eingangs- oder Ausgangsworte zu reduzieren, und<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. |
| xx89(1)(2) | EXPANSION I/O MODULE ERROR | Durch ein Erweiterungs-E/A-Modul wurde ein Fehler verursacht. | Nicht anwenderbezogen | • Überprüfen Sie den Statusfile des E/A-Moduls (IOS-File).<br>• Schlagen Sie in der Dokumentation des jeweiligen Moduls nach, um die möglichen Ursachen eines Modulfehlers zu ermitteln. |

| Fehlercode (Hex) | Meldung | Beschreibung | Fehlerklassifizierung | Empfohlene Abhilfemaßnahme |
|---|---|---|---|---|
| xx8A[1][2] | EXPANSION I/O CABLE CONFIGURATION MISMATCH ERROR | • In der Konfiguration im Anwenderprogramm wird ein Erweiterungs-E/A-Kabel aufgeführt, das nicht vorhanden ist, oder<br>• in der Konfiguration im Anwenderprogramm wird ein Erweiterungs-E/A-Kabel aufgeführt, das zwar vorhanden ist, aber nicht mit dem in der Konfiguration angegebenen Kabel übereinstimmt. | Nicht anwenderbezogen | • Korrigieren Sie das Anwenderprogramm, und löschen Sie das nicht vorhandene Kabel, und<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus, oder<br>• fügen Sie das fehlende Kabel hinzu.<br>• Spannungsversorgung aus- und wieder einschalten. |
| xx8B[1][2] | EXPANSION I/O POWER SUPPLY CONFIGURATION MISMATCH ERROR | • In der Konfiguration im Anwenderprogramm wird ein Erweiterungs-E/A-Netzteil aufgeführt, das nicht vorhanden ist, oder<br>• in der Konfiguration im Anwenderprogramm wird ein Erweiterungs-E/A-Netzteil aufgeführt, das zwar vorhanden ist, aber nicht mit dem in der Konfiguration angegebenen Netzteil übereinstimmt. | Nicht anwenderbezogen | • Korrigieren Sie das Anwenderprogramm, und löschen Sie das nicht vorhandene Netzteil, und<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus, oder<br>• fügen Sie bei ausgeschalteter Spannungsversorgung das fehlende Netzteil hinzu. |
| xx8C[1][2] | EXPANSION I/O OBJECT TYPE MISMATCH | Ein in der E/A-Konfiguration des Anwenderprogramms aufgeführtes Erweiterungs-E/A-Objekt (z. B. Kabel, Netzteil, oder Modul) stimmt nicht mit dem physischen Objekt überein. | Nicht anwenderbezogen | • Korrigieren Sie die E/A-Konfiguration im Anwenderprogramm, damit die genannten Objekttypen der tatsächlichen Konfiguration entsprechen, und<br>• Kompilieren und laden Sie das Programm erneut, und aktivieren Sie dann den RUN-Modus. Oder<br>• Passen Sie die tatsächliche Konfiguration an die E/A-Konfiguration im Anwenderprogramm an.<br>• Spannungsversorgung aus- und wieder einschalten. |
| 0x1F39 | INVALID STRING LENGTH[3] | Das erste Wort der Zeichenkettendaten enthält einen negativen Wert, eine Null oder einen Wert größer als 82. | Behebbar | Überprüfen Sie, ob das erste Wort des Zeichenkettenelements ungültige Werte enthält, und korrigieren Sie die Daten gegebenenfalls. |

(1) xx weist auf die Modulnummer hin. Wenn xx = 0 kann die Störung nicht bis zu einem bestimmten Modul rückverfolgt werden.

(2) In diesem Fehlercode bedeutet „xx", dass der Fehler an der Position des letzten korrekt konfigurierten Erweiterungs-E/A-Moduls +1 auftritt. Verwenden Sie diese Informationen in Verbindung mit dem speziellen Fehlercode, um die Fehlerquelle zu bestimmen.

(3) Gilt für MicroLogix 1500 1764-LSP-Prozessoren der Serie B und für 1764-LRP-Prozessoren.

## Unterstützung durch Rockwell Automation

Bevor Sie Unterstützung von Rockwell Automation oder des Distri- butors vor Ort anfordern, sollten Sie folgende Informationen zusammenstellen:

- Steuerungstyp, Serienbezeichnung und Revisionsbezeichnung der Basiseinheit
- Serienbezeichnung, Revisionsbezeichnung und FRN-Nummer des Prozessors (an der Unterseite der Prozessoreinheit)

  **HINWEIS** Sie finden die FRN in dem Wort S:59 (Betriebs-system-FRN) in dem Statusfile.

- Statusanzeige an Steuerungs-LED
- Fehlercodes der Steuerung (in Bit S:6 des Statusfiles)

Die Telefonnummern von Rockwell Automation finden Sie auf der Rückseite dieses Handbuchs.

Sie finden uns auch im Internet unter folgender Adresse: http://www.rockwellautomation.com.

# Protokollkonfiguration

In diesem Anhang wird die Konfiguration der Kommunikations- protokolle beschrieben. Folgende Protokolle werden von allen RS-232-Kommunikationskanälen unterstützt:

- DH-485
- DF1-Vollduplex
- DF1 Halbduplex Slave
- Modbus™ RTU Slave
- ASCII

Folgende Befehle werden in diesem Anhang beschrieben:


- „DH-485-Kommunikationsprotokoll“ auf Seite E-2
- „DF1-Vollduplex-Protokoll“ auf Seite E-5
- „DF1-Halbduplex-Protokoll“ auf Seite E-6
- „Modbus™-RTU-Slave-Protokoll (nur MicroLogix 1200-Steuerungen und MicroLogix 1500-Prozessoren der Serie B und höher)“ auf Seite E-9
- „ASCII-Treiber (nur MicroLogix 1200- und 1500-Steuerungen der Serie B und höher)“ auf Seite E-15


Informationen zu den erforderlichen Netzwerkgeräten und -zubehör-teilen finden Sie in dem Benutzerhandbuch Ihrer Steuerung.

# DH-485-Kommunikations-protokoll

In diesem Abschnitt werden die Funktionen, die Architektur und die Leistungsmerkmale des DH-485-Netzwerks beschrieben. Er erleichtert Ihnen auch die Planung und Bedienung der Steuerung auf einem DH-485-Netzwerk.

## DH-485-Netzwerkbeschreibung

Das DH-485-Protokoll definiert die Kommunikation zwischen mehreren Geräten auf einem einzelnen Aderpaar. Das DH-485-Protokoll verwendet RS-485-Halbduplex als physische Schnittstelle. (RS-485 ist eine Definition elektrischer Merkmale und kein Protokoll.) RS-485 verwendet Geräte, die in einer gemeinsamen Datenverbindung eingesetzt werden können; dies ermöglicht eine einfache gemeinsame Nutzung von Daten durch mehrere Geräte.

Merkmale des DH-485-Netzwerks:

- Verbindung von 32 Geräten
- Multi-Master-Fähigkeit
- die Zugriffssteuerung der TokenWeitergabe
- die Fähigkeit, Netzknoten hinzuzufügen oder zu entfernen, ohne dabei das Netzwerk zu stören
- maximale Netzwerklänge 1219 m

Das DH-485-Protokoll unterstützt zwei Geräteklassen: Befehlsgeber und Befehlsempfänger. Alle Befehlsgeber im Netzwerk haben die Möglichkeit, Nachrichtenübertragungen einzuleiten. Um zu bestim- men, welcher Befehlsgeber sendeberechtigt ist, wird ein Tokenweitergabe-Algorithmus verwendet.

Der folgende Abschnitt beschreibt das Protokoll, das zur Steuerung von Nachrichtenübertragungen im DH-485-Netzwerk verwendet wird.

## DH-485-Token-Rotation

Ein Netzknoten, auf dem sich der Token befindet, kann eine Nachricht an das Netzwerk senden. Jeder Knoten verfügt über eine bestimmte Anzahl von Übertragungen, sobald sich der Token auf diesem Netz- knoten befindet (entsprechend dem Token-Besitzfaktor). Nach Übertragung einer Nachricht durch einen Netzknoten wird der Knoten an das nächste Gerät weitergegeben.

Die Netzknotenadressen müssen in einem Bereich zwischen 0 und 31 liegen. Das Netzwerk muss mindestens einen Befehlsgeber enthalten (z. B. eine MicroLogix-Steuerung oder einen Prozessor SLC 5/02™ oder höher).

## DH-485-Konfigurationsparameter

Wenn die Kommunikation für ein DH-485-Netzwerk konfiguriert wird, können folgende Parameter geändert werden:

**Tabelle E.1:**

| Parameter | Optionen | Standardwert der Programmiersoftware |
|---|---|---|
| Übertragungsgeschwindigkeit | 9600, 19200 | 19200 |
| Netzknotenadresse | 1 bis 31 dezimal | 1 |
| Token-Besitzfaktor | 1 bis 4 | 2 |
| höchste Netzknotenadresse | 1 bis 31 | 31 |

Die wichtigsten Fragen zur Software, die vor der Installation eines Netzwerks geklärt werden müssen, werden in den folgenden Ab-schnitten erörtert.

## Hinweise zur Software

Die softwarebezogenen Hinweise betreffen die Konfiguration des Netzwerks und die Parameter, die auf die jeweiligen Netzwerkan-forderungen eingestellt werden können. Die folgende Auflistung enthält alle wichtigen Aspekte der Konfiguration, die einen entscheidenden Einfluss auf die Netzwerkleistung ausüben:

- Anzahl der Netzknoten im Netzwerk
- Adressen dieser Netzknoten
- Baudrate

Um eine optimale Netzwerkleistung (Geschwindigkeit) zu erzielen, werden in den folgenden Abschnitten die verschiedenen Netzwerk-aspekte erläutert und die Wahl der Parameter beschrieben. Weitere Informationen finden Sie in der Dokumentation zu der Programmiersoftware.

### *Anzahl der Netzknoten*

Die Anzahl der Netzknoten im Netzwerk hat einen direkten Einfluss auf die Geschwindigkeit der Datenübertragung zwischen einzelnen Netzknoten. Die Datenübertragungsgeschwindigkeit wird durch unnötige Netzknoten (z. B. ein zweites, nicht benutztes Programmier-gerät) beeinträchtigt. Die maximal zulässige Anzahl von Netzknoten im Netzwerk ist 32.

### *Einstellung der Netzknotenadresse*

Die maximale Netzwerkleistung wird bei sequenzieller Adressierung der Netzwerkknoten erzielt. Befehlsgebern, wie z. B. PCs, sollten die niedrigste Adressennummer zugewiesen werden, um die zur Initia-lisierung des Netzwerks erforderliche Zeit zu minimieren. Der gültige Bereich für die MicroLogixSteuerungen liegt zwischen 1 und 31 (Steuerungen können nicht die Netzknotenadresse 0 haben). Die Standardeinstellung ist 1. Die Netzknotenadresse wird in dem Kommunikations-Statusfile der Steuerung (CS0:5/0 bis CS0:5/7) gespeichert. Konfigurieren Sie die Netzknotenadresse über die Option *Channel Configuration (Kanalkonfiguration)* in RSLogix 500. Wählen Sie die Registerkarte *Kanal 0*. Die Netzknotenadresse wird als Quellenkennung angegeben.

### *Baudrate der Steuerung einstellen*

Die maximale Netzwerkleistung wird bei der höchsten Baudrate von 19 200 erzielt. Dies ist bei MicroLogix-Geräten in einem DH-485-Netzwerk die Standard-Baudrate. Alle Geräte müssen auf dieselbe Baudrate eingestellt sein. Diese Rate wird in dem Kommunikations-Statusfile der Steuerung (CS0:5/8 bis CS0:5/15) gespeichert. Konfigu-rieren Sie die Baudrate über die Option *Channel Configuration (Kanalkonfiguration)* in RSLogix 500. Wählen Sie die Registerkarte *Kanal 0*.

### *Maximale Netzknotenadresse festlegen*

Sobald das Netzwerk eingerichtet wurde und sicher ist, dass keine weiteren Geräte hinzugefügt werden sollen, können Sie die Leistung des Netzwerks durch Anpassung der maximalen Netzknotenadresse der Steuerungen steigern. Als maximale Netzknotenadresse sollte dabei die höchste verwendete Netzknotenadresse angegeben werden.

**WICHTIG** An allen Geräten sollte dieselbe maximale Netzknoten-adresse eingestellt sein.

### *Fernpaketunterstützung bei MicroLogix 1200 und 1500*

Diese Steuerungen können auf Kommunikationsanfragen und Befehle anderer, nicht aus dem zentralen DH-485-Netzwerk stammender Geräte, reagieren oder solche Anfragen und Befehle starten. Diese Eigenschaft ist vor allem bei Installationsarbeiten hilfreich, bei denen eine Kommunikation zwischen dem DH-485- und dem DH+-Netzwerk erforderlich ist.

# DF1-Vollduplex-Protokoll

Das DF1-Vollduplex-Protokoll ermöglicht eine Punkt-zu-Punkt-Verbindung zwischen zwei Geräten. Dieses Protokoll kombiniert die Datentransparenz (American National Standards Institute ANSI – Spezifikation X3.28-1976, Unterkategorie D1) mit der gleichzeitigen Zweiwegübertragung mit eingebetteten Antworten (Unterkategorie F1).

Die MicroLogix-Steuerungen unterstützen das DF1-Vollduplex-Protokoll über einen RS-232-Anschluss an externe Geräte, wie beispielsweise Computer oder andere Steuerungen, die DF1-Vollduplex unterstützen.

DF1 ist ein offenes Protokoll. Weitere Informationen finden Sie in der Allen-Bradley-Dokumentation Publikation 1770-6.5.16 *DF1 Protocol and Command Set Reference Manual.*

## Funktionsweise des DF1-Vollduplex-Protokolls

Das DF1-Vollduplex-Protokoll (oder DF1-Punkt-zu-Punkt-Protokoll) wird bei einer RS-232-Punkt-zu-Punkt-Kommunikation eingesetzt. Dieser Protokolltyp unterstützt gleichzeitige Übertragungen zwischen zwei Geräten in beiden Richtungen. DF1-Protokolle steuern die Nachrichtenübertragung, erkennen und verweisen auf Fehler und führen Wiederholungen durch, wenn Fehler festgestellt wurden.

Ist der Systemtreiber „DF1Vollduplex“, können die folgenden Parameter geändert werden:

**Tabelle E.2 Konfigurationsparameter für DF1-Vollduplex-Kommunikation**

| Parameter | Optionen | Standardwert der Programmiersoftware |
|---|---|---|
| Ubertragungsgeschwindigkeit | 300, 600, 1200, 2400, 4800, 9600, 19200, 38400 | 19200 |
| Parität | keine, gerade | keine |
| Quellen-ID (Knotenadresse) | 0 bis 254 dezimal | 1 |
| Steuerzeile | Kein Handshaking, Vollduplex-Modem-Handshaking | kein Handshaking |
| Fehlererkennung | CRC, BCC | CRC |
| Eingebettete Antworten | automatische Erkennung, aktiviert | automatische Erkennung |
| Doppelpaketerkennung | aktiviert, deaktiviert | aktiviert |
| ACK-Zeitablauf (x20 ms) | 1 bis 65535 Zählereinheiten (in Schritten von 20 ms) | 50 |
| Wiederholungen bei negativer Rückmeldung | 0 bis 255 | 3 |
| ENQ-Wiederholungen | 0 bis 255 | 3 |
| Stopp-Bits | keine Einstelloption, immer 1 | 1 |

# DF1-Halbduplex-Protokoll

Das DF1-Halbduplex-Protokoll ermöglicht ein Mehrpunktnetzwerk mit einem Master und mehreren Slaves. Das DF1-Halbduplex-Protokoll unterstützt die Datentransparenz (American National Standards Institute ANSI - Spezifikation X3.28-1976, Unterkategorie D1). Im Gegensatz zum DF1-Vollduplex-Protokoll ist die Datenüber-tragung jeweils nur in einer Richtung möglich. Der RS-232-Anschluss der MicroLogix-Steuerung kann sowohl als Halbduplex-Programmieranschluss wie auch als Halbduplex-Peer-to-Peer-Anschluss für Nachrichtenübertragung verwendet werden.

## Funktionsweise des DF1-Halbduplex-Protokolls

Das Master-Gerät startet alle Kommunikationsverbindungen durch Sendeaufrufe an die einzelnen Slave-Geräte. Die Slave-Geräte können Nachrichtenpakete nur nach Eingang eines Sendeaufrufs vom Master übertragen. Deshalb müssen in regelmäßigen Abstände Sendeaufrufe vom Master an die Slave-Geräte übertragen werden, damit diese eine Möglichkeit zur Nachrichtenübertragung haben. Für die Sendeaufruf- funktion im Master stehen zwei Konfigurationsmöglichkeiten zur Verfügung: entweder wird ein Slave-Gerät wiederholt zur Übertragung aufgefordert, bis das Slave-Gerät signalisiert, dass keine weiteren Nachrichtenpakete zur Übertragung anstehen, oder aber der Sende-aufruf wird nur einmalig übertragen.

Eine zusätzliche Funktion des DF1-Halbduplexprotokolls ist die Mög-lichkeit der Aktivierung eines MSG-Befehls durch ein Slave-Gerät in dessen eigenem Kontaktplan; auf diese Weise können Daten an andere Slave-Geräte übertragen oder von diesen angefordert werden. Bei einem Sendeaufruf an das initiierende Slave-Gerät wird der MSG-Befehl an den Master übertragen. Das Master-Gerät erkennt in diesem Fall, dass die Nachricht für ein anderes Slave-Gerät bestimmt ist und leitet die Nachricht sofort an den betreffenden Slave weiter. Diese Slave-to-Slave-Übertragung ist eine Funktion des Master-Geräts, die auch von der Programmiersoftware zum Hoch- und Herunterladen von Programmen zu/von Prozessoren in dem DF1-Halbduplex-Verbund verwendet wird.

Die MicroLogix-Steuerungen können nur als Slave-Geräte eingesetzt werden. Ein Gerät, das die Master-Funktion übernimmt, ist deshalb erforderlich. Das DF1-Halbduplex-Master-Protokoll wird von ver-schiedenen Allen-Bradley-Produkten unterstützt. Hierzu gehören die Prozessoren SLC 5/03™ und höher, erweiterte PLC-5®-Prozessoren und Rockwell Software RSLinx (Version 2.0 und höher).

Das DF1-Halbduplex-Protokoll unterstützt bis zu 255 Geräte (Adres-sen 0 bis 254), wobei Adresse 255 für Master-Rundsendungen reserviert ist. Die MicroLogix-Steuerungen unterstützen den Empfang von Rundsendungen, können jedoch keinen Rundsendebefehl initiieren. Sie unterstützen Halbduplex-Modems mit RTS/CTS-Hardware-Handshaking.

Ist der Systemtreiber „DF1-Halbduplex-Slave“, können die folgenden Parameter geändert werden:

**Tabelle E.3 DF1-Halbduplex-Slave-Konfigurationsparameter**

| Parameter | Optionen | Standardwert der Programmier-software |
|---|---|---|
| Übertragungs-geschwindigkeit | 300, 600, 1200, 2400, 4800, 9600, 19200, 38400 | 1200 |
| Parität | keine, gerade | keine |
| Quellen-ID (Knotenadresse) | 0 bis 254 dezimal | 1 |
| Steuerzeile | Kein Handshaking, Halbduplex-Modem | kein Handshaking |
| Fehlererkennung | CRC, BCC | CRC |
| EOTUnterdrückung | aktiviert, deaktiviert<br>Wenn die EOT-Unterdrückung aktiviert ist, antwortet der Slave nur auf Sendeaufrufe, wenn eine Nachricht in der Warteschlange steht. Dies führt zu einer effizienten Nutzung der Modem-Übertragungskapazität. | deaktiviert |
| Doppelpaketer-kennung | aktiviert, deaktiviert<br>Erkennt und löscht doppelte Antworten auf eine Nachricht. Wenn die Kommunikationsverbindung gestört ist und die Funktion für die Übertragungswiederholung des Absenders nicht auf 0 gesetzt ist, kann es zur Doppelübertragung von Antworten kommen. | aktiviert |
| Zeitablauf bei Sendeaufruf (x20 ms) | 0 bis 65535 (kann in Schritten von 20 ms eingestellt werden)<br>Der Zeitablauf bei Sendeaufruf wird nur wirksam, wenn ein Slave-Gerät einen MSG-Befehl initiiert. Hierbei handelt es sich um die Zeitspanne, während derer ein Slave-Gerät auf einen Sendeaufruf von einem Master-Gerät wartet. Wenn das Slave-Gerät innerhalb der angegebenen Zeit keinen Sendeaufruf erhält, wird ein MSG-Fehler erzeugt, und der MSG-Fehler muss von dem Kontaktplan erneut in die Warteschlange gestellt werden. Wenn MSG-Befehle verwendet werden, sollte der Wert für den Zeitablauf bei Sendeaufruf nicht null sein. Bei dem Wert 0 ist der Zeitablauf bei Sendeaufruf deaktiviert. | 3000 |
| RTS-Ausschaltver-zögerung (x20 ms) | 0 bis 65535 (kann in Schritten von 20 ms eingestellt werden)<br>Gibt die zeitliche Verzögerung zwischen der Übertragung des letzten seriellen Zeichens an das Modem und der Deaktivierung des RTS an. Damit erhält das Modem zusätzliche Zeit zur Übertragung des letzten Zeichens eines Pakets. | 0 |
| RTS-Sendever-zögerung (x20 ms) | 0 bis 65535 (kann in Schritten von 20 ms eingestellt werden)<br>Gibt die zeitliche Verzögerung zwischen der RTS-Aktivierung und der erstmaligen Prüfung einer CTS-Antwort an. Einzusetzen bei Modems, die nicht in der Lage sind, nach Eingang von RTS sofort mit CTS zu antworten. | 0 |
| Nachrichten-wiederholungen | 0 bis 255<br>Gibt die Anzahl der Wiederholungsversuche zur Übertragung eines Nachrichtenpakets durch ein Slave-Gerät an, wenn keine Bestätigung von dem Master-Gerät empfangen wird. Einzusetzen in störanfälligen Umgebungen, in denen Nachrichtenpakete bei der Übertragung beschädigt werden können. | 3 |
| Verzögerung vor Übertragung (x1 ms) | 0 bis 65535 (kann in Schritten von 1 ms eingestellt werden)<br>Wenn die Handshaking-Funktion *deaktiviert* wurde, ist dies die Verzögerungszeit vor der Übertragung. Erforderlich bei physischen 1761-NET-AIC Halbduplex-Netzwerken. Diese Verzögerung wird von dem 1761-NET-AIC zum Umschalten vom Übertragungs- in den Empfangsmodus benötigt.<br>Bei der Konfiguration als *DF1-Halbduplex-Modem* ist dies die Mindestverzögerung zwischen dem Empfang des letzten Zeichens eines Pakets und der RTS-Bestätigung. | 0 |

## Hinweise zur Kommunikation als DF1-Slave-Gerät in einem Mehrpunktverbund

Bei der Kommunikation zwischen der Programmiersoftware und einer Steuerung oder zwischen zwei Steuerungen über eine Slave-to-Slave-Verbindung innerhalb eines großen Mehrpunktnetzwerks können die Geräte ihre Übertragungen nur dann rechtzeitig vornehmen, wenn sie zuvor von einem DF1 Master einen Sendeaufruf erhalten haben. Mit zunehmender Anzahl von Slave-Geräten im Verbund (bis zu 254) wird auch das Zeitintervall der Sendeaufrufe an die Programmiersoftware oder die Steuerung länger. Diese Verlängerung des Zeitintervalls kann bei Verwendung niedriger Baudraten noch zunehmen.

Mit zunehmenden Zeitintervallen müssen die folgenden Werte u.U. geändert werden, um einen Kommunikationsverlust zu vermeiden:

- Programmiersoftware: Werte für Sendeaufruf-Zeitablauf und Antwortzeitablauf erhöhen
- MicroLogix-Steuerung Wert für Sendeaufruf-Zeitablauf erhöhen

## Eigentumsrechte-Zeitablauf

Wenn ein Vorgang zum Herunterladen eines Programms durch ein Softwarepaket gestartet und damit eine Kontaktplanlogik auf eine Steuerung heruntergeladen wird, erlangt die Software die Eigentums-rechte an der Steuerung. Auf diese Weise wird verhindert, dass andere Geräte während des Herunterladens Daten aus der Steuerung lesen oder in diese schreiben. Nach Abschluss des Herunterladevorgangs erlangt wieder die Steuerung die Eigentumsrechte an dem Programm, damit andere Geräte wieder mit der Steuerung kommunizieren können.

Die Steuerung löscht die Eigentumsrechte an dem Programm, wenn innerhalb des Zeitablaufs keine unterstützten Befehle von der Steue-rung mit Eigentumsrechten übertragen werden. Wenn die Eigentums-rechte an dem Programm nach Abschluss eines Herunterladevorgangs nicht gelöscht würden, könnte die Steuerung keine Befehle von anderen Geräten annehmen, da diesen anderen Geräten nach wie vor angezeigt würde, dass ein anderes Gerät die Programmeigentumsrechte hält.

**WICHTIG** Bei der Unterbrechung eines Herunterladevorgangs, z. B. wegen elektromagnetischer Interferenz oder anderer Ereignisse, sollten Sie die Kommunikation mit der Steuerung während der Eigentumsrechte-Zeitablauf-Periode unterbrechen und dann den Herunterladevorgang neu starten. Die Eigentums-rechte-Zeitablauf-Periode beträgt 60 Sekunden. Nach dem Zeitablauf kann die Kommunikation mit der Steuerung wieder aufgenommen und der Herunterladevorgang wieder-holt werden. Die einzige andere Möglichkeit zum Löschen der Eigentumsrechte an einem Programm bestünde im Aus- und erneuten Einschalten der Spannungsversorgung der Steuerung.

## Modbus™-RTU-Slave-Protokoll (nur MicroLogix 1200-Steuerungen und MicroLogix 1500-Prozessoren der Serie B und höher)

In diesem Abschnitt werden die Konfigurationsparameter für das Modbus-RTU-Slave-Protokoll (*Remote Terminal Unit*-Übertragungs-modus) beschrieben. Weitere Informationen zu dem Modbus Slave-Protokoll finden Sie in der Spezifikation des Modbus-Protokolls (unter http://www.modbus.org).

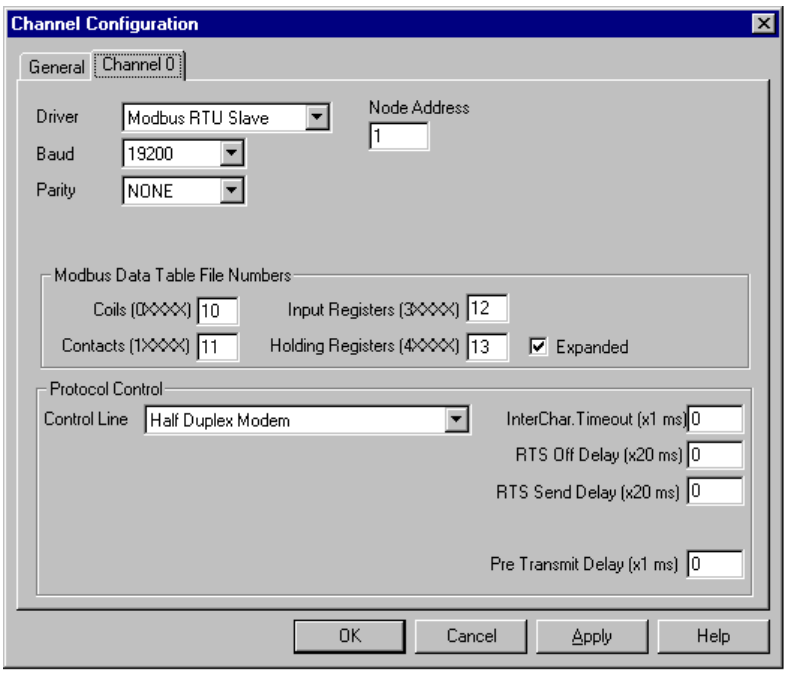
Der Modbus RTU Slave-Treiber ordnet die vier Modbus-Datentypen – Spulen, Kontakte, Eingangsregister und Halteregister – vier binären und/oder ganzzahligen Datentabellenfiles zu, die vom Anwender erzeugt wurden.

Der Spulen- und der Kontaktfile enthalten pro Register bis zu 4096 Spulen oder Kontakte, sofern der Datentafelfile auf eine maximale Größe von 256 Wörtern konfiguriert wurde. Der Eingangs-register- und der Speicherregisterfile enthalten bis zu 256 Register, sofern der Datentafelfile auf eine maximale Größe von 256 Wörtern konfiguriert wurde. Wenn die Option *Erweitert* aktiviert ist, können die Steuerungen so konfiguriert werden, dass bis zu sechs 256-Wort-Datentafelfiles für insgesamt 1536 Modbus-Halteregister verwendet werden.

**HINWEIS** Es ist möglich, einen Zugriff auf eine Gruppe an Halte-registern, die sich über zwei Files erstrecken, anzufordern. Beachten Sie, dass es aufgrund der maximalen Anzahl an Registern in einem Befehl nicht möglich ist, während eines einzelnen Modbus-Befehls auf mehr als zwei Files zuzgreifen.

## Modbus-Konfiguration

Im Nachfolgenden werden der Konfigurationsbildschirm für Modbus und der Konfigurationsvorgang näher beschrieben:




1. Wählen Sie die Registerkarte zur Konfiguration von Kanal 0, um Kanal 0 und Datenfiles für die Modbus-Kommunikation einzu-richten. *Nur 1764-LRP: Auswahl von Kanal 0 oder Kanal 1 möglich.*
2. Wählen Sie den Modbus RTU Slave-Treiber und weisen Sie die Treibermerkmale zu.
3. Geben Sie die Modbus-Datentafel-Filenummer ein. Wählen Sie das Kontrollkästchen *Erweitert*, um mehrere Datenfiles zu verwenden. *(Nur MicroLogix 1200 Serie C FRN6 und höher sowie MicroLogix 1500 Serie C FRN7 und höher. Zum Programmieren ist RSLogix 500 Version 5.50 oder höher erforderlich.)*

   **HINWEIS** Die Standardeinstellung der Steuerung ist ein Datenfile mit 256 Registern. Über das Kontroll-kästchen *Erweitert* werden zusätzliche fünf Files und 1280 Register zur Verfügung gestellt.

   Die fünf zusätzlichen Tafeln müssen nicht einzeln bestimmt werden; sie folgen der ersten Ganzzahl oder dem ersten Bitfile. Wenn der erste File beispielsweise N10 ist (oder B10), sind die zusätzlichen fünf Files N11 (oder B11), N12 (B12), N13 (B13), N14 (B14) und N15 (B15).
4. Geben Sie für jeden File die Datentafelgröße und den -typ ein. Die Datentafelfiles werden automatisch erstellt.

## Modbus-Speicherbelegung

Eine Übersicht zur Modbus-Speicherbelegung finden Sie in Tabelle E.4; eine ausführliche Beschreibung der Speicherbelegung finden Sie in Tabelle E.5:

**Tabelle E.4 Modbus-zu-MicroLogix-Speicherbelegung – Zusammenfassung**
*(nur MicroLogix 1200-Steuerungen und MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren)*

| Modbus-Adressierung | Beschreibung | Gültige MicroLogix-Adressierung | | |
|---|---|---|---|---|
| | | Filetyp | Datenfilenummer | Adresse |
| 0001 bis 4096 | Modbus-Spulen, Schreib-/Lese-Datenbereich | Bit (B) oder Ganzzahl (N) | 3 bis 255 | Bits 0 bis 4095 |
| 10001 bis 14096 | Modbus-Kontakte, Nur-Lesen-Datenbereich | Bit (B) oder Ganzzahl (N) | 3 bis 255 | Bits 0 bis 4095 |
| 30001 bis 30256 | Modbus-Eingangsregister, Nur-Lesen-Datenbereich | Bit (B) oder Ganzzahl (N) | 3 bis 255 | Wörter 0 bis 255 |
| 30501 bis 30532 | Modbus-Kommunikationsparameter | Kommunikations-Status-File | - | - |
| 31501 bis 31566 | System-Statusfile, Nur-Lesen | Status (S) | 2 | Wörter 0 bis 65 |
| 40001 bis 40256 | Modbus-Speicherregister, Schreib-/Lese-Datenbereich | Bit (B) oder Ganzzahl (N) | 3 bis 255 | Wörter 0 bis 255 |
| 40257 bis 41280(1) | Modbus-Speicherregister, Schreib-/Lese-Datenbereich | Bit (B) oder Ganzzahl (N) | 4 bis 255 | Wörter 0 bis 255 von vier Halteregister-Files |
| 41501 bis 41566 | System-Statusfile, Lesen/Schreiben | Status (S) | 2 | Wörter 0 bis 65 |
| 41793 bis 42048(1) | Modbus-Speicherregister, Schreib-/Lese-Datenbereich | Bit (B) oder Ganzzahl (N) | 8 bis 255 | Wörter 0 bis 255 von dem letzten Halteregister-File |

(1) Diese Adressen werden nur aktiv, wenn sie besonders konfiguriert werden.

**Tabelle E.5 Modbus-zu-MicroLogix-Speicherbelegung – Ausführliche Beschreibung**
*(nur MicroLogix 1200-Steuerungen und MicroLogix 1500 1764-LSP-Prozessoren der Series B und 1764-LRP-Prozessoren)*

| Modbus-Adressierung | Modbus-Adressreferenz | Modbus-Funktionscode (dezimal) |
|---|---|---|
| 0001 bis 4096 | Modbus-Spulen, Schreib-/Lese-Datenbereich | 1, 5, 15 |
| 10001 bis 14096 | Modbus-Kontakte, Nur-Lesen-Datenbereich | 2 |
| 30001 bis 30256 | Modbus-Eingangsregister, Lesen | 4 |
| 30501 | Modbus-Spulen, Datentafelfilenummer | 4 |
| 30502 | Modbus-Kontakte, Datentafelfilenummer | 4 |
| 30503 | Modbus-Eingangsregister, Datentafelfilenummer | 4 |
| 30504 | Modbus-Speicherregister, Datentafelfilenummer | 4 |
| 30506 | Verzögerung vor Senden | 4 |
| 30507 | Modbus-Slave-Adresse | 4 |
| 30508 | Zeitablauf zwischen Zeichen | 4 |
| 30509 | RTS-Sendeverzögerung | 4 |
| 30510 | RTS-Aus-Verzögerung | 4 |
| 30511 | Parität | 4 |
| 30512 | Fehlercode Darstellungsschicht | 4 |
| 30512 | Fehlercode Darstellungsschicht | 4 |
| 30513 | Fehlerzähler Darstellungsschicht | 4 |
| 30514 | Fehlercode Ausführungsfunktion | 4 |
| 30515 | Letzter übertragener Ablaufunterbrechungscode | 4 |
| 30516 | Filenummer der Fehleraufforderung | 4 |
| 30517 | Elementnummer der Fehleraufforderung | 4 |
| 30518 | Funktionscode 1 Nachrichtenzähler - Einzelausgang Spule lesen | 4 |
| 30519 | Funktionscode 2 Nachrichtenzähler - Diskretes Eingangsdaten lesen | 4 |
| 30520 | Funktionscode 3 Nachrichtenzähler - Einzelspeicherregister lesen | 4 |
| 30521 | Funktionscode 4 Nachrichtenzähler - Einzeleingangsregister lesen | 4 |
| 30522 | Funktionscode 5 Nachrichtenzähler - Einzelausgang Spule setzen/löschen | 4 |
| 30523 | Funktionscode 6 Nachrichtenzähler - Einzelspeicherregister lesen/schreiben | 4 |
| 30524 | Funktionscode 8 Nachrichtenzähler - Diagnose durchführen | 4 |
| 30525 | Funktionscode 15 Nachrichtenzähler - Ausgangsspulenblock setzen/löschen | 4 |
| 30526 | Funktionscode 16 Nachrichtenzähler - Speicherregisterblock lesen/schreiben | 4 |
| 30527 | Modemstatus | 4 |
| 30528 | Summe der beantworteten Nachrichten für diesen Slave | 4 |
| 30529 | Summe der Nachrichten an diesen Slave | 4 |
| 30530 | Summe der angezeigten Nachrichten | 4 |
| 30531 | Fehlerzähler Netzwerkebene | 4 |
| 30532 | Fehler Netzwerkebene | 4 |
| 31501 bis 31566 | System-Statusfile, Nur-Lesen | 4 |
| 40001 bis 40256 | Modbus-Halteregister, Lese-/Schreib-Datenbereich (erster Halteregiste-File). | 3, 6, 16 |
| 40257 bis 40512 | Modbus-Halteregister, Lese-/Schreib-Datenbereich (zweiter Halteregiste-File). | 3, 6, 16 |
| 40513 bis 40768 | Modbus-Halteregister, Lese-/Schreib-Datenbereich (dritter Halteregiste-File). | 3, 6, 16 |
| 40769 bis 41024 | Modbus-Halteregister, Lese-/Schreib-Datenbereich (vierter Halteregiste-File). | 3, 6, 16 |
| 41025 bis 41280 | Modbus-Halteregister, Lese-/Schreib-Datenbereich (fünfter Halteregiste-File). | 3, 6, 16 |
| 41501 bis 41566 | System-Statusfile, Lesen/Schreiben | 3, 6, 16 |
| 41793 bis 42048 | Modbus-Halteregister, Lese-/Schreib-Datenbereich (sechster Halteregiste-File). | 3, 6, 16 |

Die Steuerung reagiert auf die in Tabelle E.6 aufgeführten Modbus-Befehlsfunktionscodes:

**Tabelle E.6 Unterstützte Modbus-Befehle**
*(MicroLogix 1200-Steuerungen, MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren)*

| Befehl | Funktionscode (dezimal) | Subfunktionscode (dezimal) |
|---|---|---|
| Spulenstatus lesen | 1 | - |
| Eingangsstatus lesen | 2 | - |
| Speicherregister lesen | 3 | - |
| Eingangsregister lesen | 4 | - |
| Einzelspule setzen und rücksetzen(1) | 5 | - |
| Einzelhalteregister schreiben(1) | 6 | - |
| Befehlsdatenecho | 8 | 0 |
| Diagnosezähler löschen | 8 | 10 |
| Mehrere Spulen setzen und rücksetzen(1) | 15 | - |
| Mehrere Halteregister schreiben(1) | 16 | - |

(1) Broadcast wird für diesen Befehl unterstützt.

Beim Empfang eines nicht unterstützten oder falsch formatierten Modbus-Befehls antwortet die Steuerung mit einem der in Tabelle E.7 aufgeführten Ausnahmecodes:

**Tabelle E.7 Modbus-Fehlercodes**
*(MicroLogix 1200-Steuerungen, MicroLogix 1500 1764-LSP-Prozessoren der Serie B und 1764-LRP-Prozessoren)*

| Fehlercode | Fehler | Beschreibung | Übertragener Ausnahmecode(1) |
|---|---|---|---|
| 0 | Kein Fehler. | | keine |
| 1 | Funktionscode ohne Rundsendung. | Die Funktion unterstützt keine Rundsendung. | keine Übertragung |
| 2 | Funktionscode nicht unterstützt | Diese Modbus-Funktion oder -Subfunktion wird nicht von der Steuerung unterstützt. | 1 |
| 3 | Ungültige Befehlslänge | Die Länge des Modbus-Befehls ist ungültig. | 3 |
| 4 | Ungültige Länge | Die Funktion versucht, über das Ende des Datenfiles hinaus zu lesen/schreiben. | 3 |
| 5 | Ungültiger Parameter | Die Funktion kann mit diesen Parametern nicht ausgeführt werden. | 1 |
| 6 | Ungültiger Filetyp | Die Filenummer, auf die verwiesen wird, weist nicht den richtigen Filetyp auf. | 2 |
| 7 | Ungültige Filenummer | Die Filenummer ist nicht vorhanden. | 2 |
| 8 | Ungültige Modbus-Adresse | Die Funktion versucht, auf eine ungültige Modbus-Adresse zuzugreifen.(2) | 3 |
| 9 | Tafel schreibgeschützt | Die Funktion versucht, in einen schreibgeschützten File zu schreiben. | 3 |
| 10 | Kein Filezugriff | Auf diesen File besteht kein Zugriff. | 2 |
| 11 | Eigentumsrechte an File bereits vergeben | Ein anderer Prozess hält bereits die Eigentumsrechte an dem File. | 2 |

(1) Wenn der Modbus-Befehl mit einer gültigen Broadcast-Adresse gesendet wurde, wird bei den Fehlercodes 2 bis 11 keine Ablaufunterbrechungsantwort generiert.

(2) Siehe Tabelle E.4 auf Seite E-11 für gültige Modbus-Speicherbelegung.

Ist der Systemtreiber „Modbus-RTU-Slave“, können die folgenden Parameter des Kommunikationsanschlusses geändert werden:

**Tabelle E.8 Konfigurationsparameter für Modbus-RTU-Slave-Kommunikation**
*(nur MicroLogix 1200-Steuerungen und MicroLogix 1500-Prozessoren der Serie B und höher)*

| Parameter | Optionen | Standardwert der Programmier-software |
|---|---|---|
| Übertragungs-geschwindigkeit | 300, 600, 1200, 2400, 4800, 9600, 19200, 38400 | 19200 |
| Parität | keine, gerade, ungerade | keine |
| Netzknotenadresse | 1 bis 247 dezimal | 1 |
| Steuerzeile | Kein Handshaking, Halbduplex-Modem | kein Handshaking |
| Zeitablauf zwischen Zeichen (x1 ms) | 0 bis 6553 (kann in Schritten von 1 ms eingestellt werden) 0 = 3,5 Zeichenzeiten<br>Gibt die Mindestverzögerung zwischen Zeichen an, die das Ende eines Nachrichtenpakets signalisiert. | 0 |
| Zuordnung der Modbus-Daten-tafelfile-nummern | Spulen (diskrete Ausgänge, Modbus-Adressen 0001 bis 4096) Bereich = 3 bis 255, 0 = kein File | 0 |
| | Kontakte (diskrete Eingänge, Modbus-Adressen 10001 bis 14096) Bereich = 3 bis 255, 0 = kein File | 0 |
| | Eingangsregister (nur Lesen, Modbus-Adressen 30001 bis 30256) Bereich = 3 bis 255, 0 = kein File | 0 |
| | Speicherregister (Lesen/Schreiben, Modbus-Adressen 40001 bis 40256) Bereich = 3 bis 255, 0 = kein File | 0 |
| RTS-Ausschalt-verzögerung (x20 ms) | 0 bis 65535 (kann in Schritten von 20 ms eingestellt werden)<br>Gibt die zeitliche Verzögerung zwischen der Übertragung des letzten seriellen Zeichens an das Modem und der Deaktivierung des RTS an. Damit erhält das Modem zusätzliche Zeit zur Übertragung des letzten Zeichens eines Pakets. | 0 |
| RTS-Sendeverzö-gerung (x20 ms) | 0 bis 65535 (kann in Schritten von 20 ms eingestellt werden)<br>Gibt die zeitliche Verzögerung zwischen der RTS-Aktivierung und der erstmaligen Prüfung einer CTS-Antwort an. Einzusetzen bei Modems, die nicht in der Lage sind, nach Eingang von RTS sofort mit CTS zu antworten. | 0 |
| Verzögerung vor Übertragung (x1 ms) | 0 bis 65535 (kann in Schritten von 1 ms eingestellt werden)<br>Wenn die Handshaking-Funktion *deaktiviert* wurde, ist dies die Verzögerungszeit vor der Übertragung. Erforderlich bei physischen 1761-NET-AIC Halbduplex-Netzwerken. Diese Verzögerung wird von dem 1761-NET-AIC zum Umschalten vom Übertragungs- in den Empfangsmodus benötigt.<br>Bei der Konfiguration als *DF1-Halbduplex-Modem* ist dies die Mindestverzögerung zwischen dem Empfang des letzten Zeichens eines Pakets und der RTS-Bestätigung. | 0 |

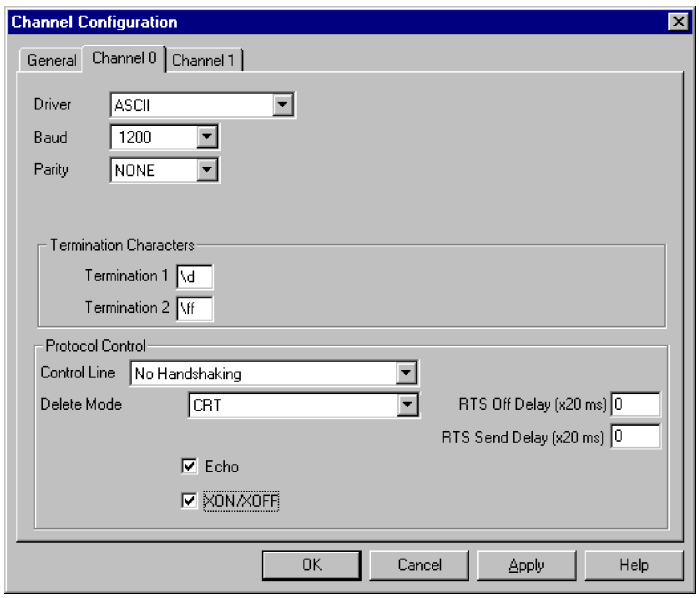
## ASCII-Treiber (nur MicroLogix 1200- und 1500-Steuerungen der Serie B und höher)

Der ASCII-Treiber ermöglicht den Anschluss an andere ASCII-Geräte, wie Strichcodeleser, Waagen, serielle Drucker und andere intelligente Geräte.

Das ASCII-Protokoll kann verwendet werden, wenn am RS-232-Anschluss für den ASCII-Treiber Kanal 0 konfiguriert wird (*nur für den 1764-LRP-Prozessor kann Kanal 0 oder Kanal 1 ausgewählt werden*). Wurde der Anschluss für ASCII konfiguriert, werden alle empfangenen Daten in einem Pufferspeicher abgelegt. Zum Zugriff auf die Daten verwenden Sie bitte die ASCII-Befehle in Ihrem Kontaktplanlogik-Programm. Weitere Informationen zur Verwendung der ASCII-Befehle finden Sie im Abschnitt „ASCII-Befehle“ auf Seite 20-1. Außerdem können Sie die ASCII-Zeichenkettendaten an die meisten angeschlossenen Geräte senden, die ASCII-Daten/Zeichen akzep-tieren.

**HINWEIS** Die ASCII-Befehle können nur verwendet werden, wenn ein Kanal für ASCII konfiguriert wurde. Wird ein Nachrichtenbefehl (MSG-Befehl) verwendet, der auf den Kanal verweist, tritt ein Fehler auf.

Im Folgenden ist der Kanalkonfigurationsbildschirm abgebildet:




Die Steuerung aktualisiert Änderungen an der Kanalkonfiguration beim nächsten Ausführen eines SVC-Befehls, REF-Befehls oder beim Durchführen der Kommunikationsbearbeitung (je nachdem, was zuerst auftritt).

Ist der Treiber für ASCII konfiguriert, können folgende Parameter geändert werden:

**Tabelle E.9 Konfigurationsparameter für ASCII-Kanal**

| Parameter | Beschreibung | Standardwert der Programmiersoftware |
|---|---|---|
| Übertragungs-geschwindigkeit | Umschalten zwischen den Kommunikationsraten 300, 600, 1200, 2400, 4800, 9600, 19200 und 38400. | 1200 |
| Parität | Umschalten zwischen den Paritätswerten „None" (Keine), „Odd" (Ungerade) und „Even" (Gerade). | Keine |
| Termination 1 (Abschluss 1) | Angeben des ersten Abschlusszeichens. Das Abschlusszeichen definiert die aus ein bzw. zwei Zeichen bestehende Zeichenfolge, die das Ende einer empfangenen ASCII-Zeile angibt. Durch Setzen des ersten ASCII-Abschlusszeichens auf undefiniert (\ff) gibt der Anwender an, dass kein Abschluss für eine empfangene ASCII-Zeile verwendet wird. | \d |
| Termination 2 (Abschluss 2) | Angeben des zweiten Abschlusszeichens. Das Abschlusszeichen definiert die aus ein bzw. zwei Zeichen bestehende Zeichenfolge, die das Ende einer empfangenen ASCII-Zeile angibt. Durch Setzen des zweiten ASCII-Abschlusszeichens auf undefiniert (\ff) und des ersten ASCII-Abschlusszeichens auf einen definierten Wert (\d) gibt der Anwender eine aus einem Zeichen bestehende Abschlusszeichenfolge an. | \ff |
| Steuerzeile | Umschalten zwischen den Einstellungen „No Handshaking" (Kein Handshaking), „Half-Duplex Modem" (Halbduplex-Modem) und „Full-Duplex Modem" (Vollduplex-Modem) | Kein Handshaking |
| Delete Mode (Lösch-Modus) | Über diesen Parameter können Sie den Modus des „Lösch"-Zeichens auswählen. Umschalten zwischen „Ignore" (Ignorieren), „CRT" und „Printer" (Drucker).<br>Der Lösch-Modus wirkt sich auf die an das dezentrale Gerät zurückgegebenen Zeichen aus. Ist der Lösch-Modus aktiviert, wird das vorherige Zeichen aus dem Empfangspufferspeicher entfernt.<br>• Im CRT-Modus gibt die Steuerung beim Erkennen eines Lösch-Zeichens an das Gerät drei Zeichen zurück: Rückschritt, Leerzeichen und Rückschritt. Auf diese Weise wird das vorherige Zeichen im Terminal gelöscht.<br>• Im Drucker-Modus gibt die Steuerung beim Erkennen eines Lösch-Zeichens zunächst das Schrägstrich-Zeichen und anschließend das gelöschte Zeichen zurück.<br>Aktivieren Sie den Parameter „Echo", um den Lösch-Modus verwenden zu können. | Ignore (Ignorieren) |
| Echo | Ist der Echo-Modus aktiviert, werden alle empfangenen Zeichen an das dezentrale Gerät zurückgegeben. Dies ermöglicht Ihnen das Anzeigen der Zeichen auf einem an der Steuerung angeschlossenen Terminal. Umschalten zwischen „Enabled" (Aktiviert) und „Disabled" (Deaktiviert). | Deaktiviert |
| XON/XOFF | Ermöglicht Ihnen das Aktivieren oder Deaktivieren des XON/XOFF-Software-Handshaking. Für das XON/XOFF-Software-Handshaking sind die Steuerzeichen XON und XOFF im ASCII-Zeichensatz erforderlich.<br>Wenn der Empfänger das XOFF-Zeichen empfängt, unterbricht der Transmitter die Übertragung, bis der Empfänger das Zeichen XON empfangen hat. Hat der Empfänger das Zeichen XON nach 60 Sekunden noch nicht empfangen, setzt der Transmitter die Übertragung der Zeichen automatisch fort..<br>Ist dagegen der Empfängerpufferspeicher zu über 80 % belegt, wird an das dezentrale Gerät ein XOFF-Zeichen gesendet, um die Übertragung zu unterbrechen. Ist danach der Empfängerpuffer-speicher zu weniger als 80 % belegt, wird an das dezentrale Gerät ein XON-Zeichen gesendet, um die Übertragung wieder aufzunehmen. | Deaktiviert |
| RTS-Ausschalt-verzögerung (x20 ms) | Ermöglicht das Festlegen der Verzögerung zwischen dem Beenden einer Übertragung und dem Absetzen einer Sendeaufforderung (RTS). Geben Sie den Wert für diesen Parameter in Inkrementen von 20 ms an. Die Werte müssen zwischen 0 und 65535 liegen. | 0 |
| RTS-Sendeverzö-gerung (x20 ms) | Ermöglicht das Festlegen der Verzögerung zwischen dem Absetzen der Sendeaufforderung (RTS) und dem Beginn der Übertragung. Geben Sie den Wert für diesen Parameter in Inkrementen von 20 ms an. Die Werte müssen zwischen 0 und 65535 liegen. | 0 |

Die folgenden Begriffe werden in dem vorliegenden Handbuch verwendet. Eine vollständige Auflistung der bei Allen-Bradley verwendeten Fachbegriffe finden Sie im Handbuch *Allen-Bradley Industrial Automation Glossary*, Publikationsnummer AG-7.1.

**Abfrage**
Die Abfrage besteht aus vier Elementen: Eingangsabfrage, Programmabfrage, Ausgangsabfrage und Verwaltung.

**Abfragezeit**
Die Zeit, die zur Durchführung einer Abfrage erforderlich ist.

**Adresse**
Eine Zeichenfolge, die einen Speicherstandort eindeutig kennzeichnet. Beispiel: I:1/0 ist die Speicheradresse von Daten, die sich im Eingangsfile in Wort 1, Bit 0, befinden.

**AIC+ Erweiterter Schnittstellenwandler**
Ein Gerät, das die RS-232-Isolierung zu einer RS-485-Halbduplex-Kommunikationsverbindung sicherstellt. (Bestellnummer 1761-NET-AIC.)

**Anwendung**
1) Eine Maschine oder ein Prozess, die/der von einer Steuerung gesteuert wird. 2) Die Verwendung computer- oder prozessor- gestützter Routinen für bestimmte Zwecke.

**ASCII**
American Standard Code for Information Interchange. Ein Standard zur Definition von Codes zum Informationsaustausch zwischen Geräten verschiedener Hersteller. Die Basis der Zeichensätze, die in den meisten Mikrocomputern verwendet werden. Die Zeichen wer- den jeweils durch eine Zeichenkette von sieben binären Ziffern dargestellt.

**Ausführungsmodus**
Jeder RUN- oder TEST-Modus.

**Ausgangsabfrage**
Die Steuerung schaltet die an die Ausgangsklemmen angeschlossenen Geräte ein oder aus oder modifiziert diese.

**Ausgangsgerät**
Ein Gerät, z. B. eine Kontrollleuchte oder eine Motorstarterspule, das ein Signal oder einen Befehl von der Steuerung empfängt.

**Ausschaltverzögerung**
Die Ausschaltverzögerung ist ein Maß für die Zeit, die vergeht, bis die Steuerungslogik erkannt, dass ein Signal an der Eingangsklemme der Steuerung gelöscht wurde. Diese Zeitspanne hängt von der Verzö- gerung der Schaltungskomponenten und den gewählten Filterein- stellungen ab.

**Baudrate**
Die Geschwindigkeit der Kommunikation zwischen Geräten. Die Baudrate wird normalerweise in *kBaud* angegeben. Beispielsweise bedeutet 19200 kBaud eine Baudrate von 19200 Bit pro Sekunde.

**Befehl**
Eine Mnemonik, die eine durch den Prozessor auszuführende Operation definiert. Ein Strompfad in einem Programm besteht aus einer Reihe von Eingangs- und Ausgangsbefehlen. Die Eingangsbe- fehle werden von der Steuerung geprüft und als wahr oder unwahr erkannt. Daraufhin stuft die Steuerung die Ausgangsbefehle als wahr oder unwahr ein.

**Befehlssatz**
Die Befehle, die in einer Steuerung verfügbar sind.

**Betriebsspannung**
Bei Eingängen der Spannungsbereich, in dem der Eingang einge- schaltet ist. Bei Ausgängen der zulässige Bereich für die durch den Anwender bereitgestellte Spannung.

**Bit**
Die kleinste Speichereinheit, die in der diskreten oder binären Logik verwendet wird; dabei steht der Wert 1 für EIN und der Wert 0 für AUS.

**Blockdiagramm**
Eine Methode zur Erläuterung der Logikkomponenten oder einer Sequenz von Ereignissen.

**Boolesche Operanden**
Die logischen Operanden wie UND, ODER, NICHT UND, WEDER-NOCH, NICHT und EXKLUSIV ODER, die einzeln oder in Kombination als logische Aussage oder Schaltung eingesetzt werden können. Die Ausgangsantwort zu diesen Operanden ist WAHR oder UNWAHR.

**CPU (Central Processing Unit)**
Der zentrale Teil einer speicherprogrammierbaren Steuerung, in dem Entscheidungen getroffen und Daten gespeichert werden.

**Datentafel**
Der Teil des Prozessorspeichers, der den E/A-Status und Files zur Überwachung, Bearbeitung und Veränderung von Anwenderdaten (z. B. Bits, Ganzzahlen, Zeitwerke und Zähler) enthält.

**DIN-Schiene**
Hergestellt gemäß den DIN-Standards (Deutsche Industrie-Normenausschuss); eine Metallschiene als Hilfsmittel bei der Installation und Bestückung einer Steuerung.

**DTE**
Datenendeinrichtung

**Durchsatz**
Die Zeitspanne zwischen dem Einschalten eines Eingangs und dem Ein- oder Ausschalten des entsprechenden Ausgangs. Der Durchsatz umfasst Eingangsverzögerungen, Programmabfragen, Ausgangsverzö- gerungen und den Overhead.

**E/A**
Eingang und Ausgang

**Eingangsabfrage**
Die Steuerung liest alle Eingangsgeräte, die an die Eingangsterminals angeschlossen sind.

**Eingangsgerät**
Ein Gerät, z. B. ein Drucktaster oder Schalter, das ein elektrisches Signal an die Steuerung liefert.

**Einschaltstrom**
Der vorübergehende Stoßstrom, der beim erstmaligen Anlegen von Spannung an einem Gerät oder einer Schaltung erzeugt wird.

**Einschaltverzögerung**
Die Einschaltverzögerung ist ein Maß für die Zeit, die vergeht, bis die Steuerungslogik erkannt, dass ein Signal an der Eingangsklemme der Steuerung vorliegt.

**Einzelimpuls**
Ein Programmierverfahren, bei dem ein Bit für nur eine Programm- abfrage auf EIN oder AUS gesetzt wird.

**EMI**
Elektromagnetische Interferenz

**Erweiterungs-E/A**
Die Erweiterungs-E/A ist die E/A, die über einen Bus oder ein Kabel an die Steuerung angeschlossen wird. MicroLogix 1200-Steuerungen verwenden Bulletin 1762-Erweiterungs-E/A. MicroLogix 1500- Steue- rungen verwenden Bulletin 1769-Erweiterungs-E/A. Bei MicroLogix-Steuerungen ist die gesamte E/A an Steckplatz 1 und höher die Erweiterungs-E/A.

**Festplatte**
Ein Speichergerät in einem PC.

**FET**
Field Effect Transistor. DC-Ausgang für Hochgeschwindigkeitsbetrieb.

**FIFO (First-In-First-Out)**
Die Reihenfolge, in der Daten zur Speicherung in eine Datei einge- geben und aus ihr abgerufen werden.

**File**
Daten oder Logik, die in Gruppen angeordnet sind.

**Halbduplex**
Ein Kommunikationsmodus, bei dem die Datenübertragung jeweils nur in einer Richtung möglich ist.

**Hauptsteuerrelais (MCR)**
Ein festverdrahtetes Relais, dessen Spannungsversorgung durch seriell angeschlossene Not-Ausschalter unterbrochen werden kann.

**Herunterladen**
Die Übertragung eines Programms oder Datenfiles auf ein Gerät.

**Hochladen**
Übertragung von Daten von der Steuerung an ein Programmier- oder Speichergerät.

**Integrierte E/A**
Die integrierte E/A ist die E/A auf der Steuerungsplatine. Bei MicroLogix-Steuerungen ist die gesamte E/A an Steckplatz 0 die integrierte E/A.

**Klemme**
Eine Stelle an einem E/A-Modul, mit der externe E/A-Geräte, z. B. ein Drucktaster oder eine Kontrollleuchte, verdrahtet sind.

**Kommunikationsabfrage**
Ein Teil des Arbeitszyklus der Steuerung. Die Kommunikation mit Geräten (z. B. anderen Steuerungen und Bedienerschnittstellen- geräten) erfolgt während dieser Phase.

**Kontaktplanlogik**
Ein grafisches Programmformat, das einem Leiterdiagramm ähnelt. Die Prgrammiersprache für die Kontaktplanlogik ist die am weitesten verbreitete Programmiersprache für speicherprogrammierbare Steuerungen.

**Leckstrom im AUS-Zustand**
Wenn ein mechanischer Schalter geöffnet wird (AUS-Zustand), fließt kein Strom durch diesen Schalter. Bei Halbleiterschaltern und Überspannungsschutzkomponenten, die manchmal zum Schutz von Schaltern eingesetzt werden, weisen einen geringen Stromfluss auf, wenn sich der Schalter im AUS-Zustand befindet. Dieser Strom wird als Leckstrom im AUS-Zustand bezeichnet. Ein zuverlässiger System- betrieb ist nur möglich, wenn der nominale Leckstrom im AUS-Zustand kleiner ist als der kleinste nominale Betriebsstrom des Geräts, das angeschlossen ist.

**LED (Leuchtdiode)**
Wird als Statusanzeiger für Prozessorfunktionen und Ein- und Ausgänge verwendet.

**Lesen**
Daten erfassen. Beispielsweise liest der Prozessor Informationen von anderen Geräten über eine Lesenachricht.

**LIFO (Last-In-First-Out)**
Die Reihenfolge, in der Daten zur Speicherung in eine Datei eingegeben und aus ihr abgerufen werden.

**Logik**
Allgemeine Bezeichnung für digitale Schaltungen oder programmierte Befehle zur Durchführung von Entscheidungs- und Berechnungs- funktionen.

**Mnemonik**
Ein einfacher und einprägsamer Begriff, der zur Darstellung komplexer oder umfassender Informationen verwendet wird.

**Modbus™-RTU-Slave**
Ein serielles Halbduplex-Kommunikationsprotokoll.

**Modem**
Modulator/Demodulator. Gerät, das eine Verbindung zwischen einer Datenendeinrichtung und einer Kommunikationsleitung herstellt.

**Modi**
Ausgewählte Betriebsarten. Beispiel: Run, Test oder Programm.

**Negative Logik**
Die Verwendung der binären Logik in der Weise, dass „0“ das gewünschte Spannungsniveau darstellt.

**Netzwerk**
Eine Reihe von Stationen (Netzknoten), die durch ein Kommunikationsmedium miteinander verbunden sind. Ein Netz kann aus einer einzelnen oder auch aus mehreren Verbindungen bestehen.

**Niederwertigstes Bit (LSB)**
Das Element (oder Bit) in einem Binärwort, das den kleinsten Wert aufweist.

**Nominaler Eingangsstrom**
Der Strom, der in der Regel bei der nominalen Eingangsspannung auftritt.

**Oberes Byte**
Die Bits 8 bis 15 eines Worts.

**Offline**
Wenn ein Gerät keine Abfrage/Steuerungsaufgaben durchführt oder wenn ein Programmiergerät nicht mit der Steuerung kommuniziert.

**Öffnungskontakte**
Kontakte eines Relais oder Schalters, die geschlossen werden, wenn die Spannungsversorgung des Relais unterbrochen oder der Schalter deaktiviert wird. Die Kontakte sind geöffnet, wenn am Relais Span- nung angelegt oder der Schalter aktiviert wurde.

**Offset**
Die dauerhafte Abweichung einer geregelten Variablen von einem festen Punkt.

**Online**
Wenn ein Gerät eine Abfrage/Steuerungsaufgaben durchführt oder wenn ein Programmiergerät mit der Steuerung kommuniziert.

**PCCC**
Programmable Controller Communications Commands – Kommunikationsbefehle für speicherprogrammierbare Steuerungen.

**Programmabfrage**
Ein Teil des Arbeitszyklus der Steuerung. Während der Programm- abfrage wird das Logikprogramm verarbeitet und die Ausgangsdaten aktualisiert.

**Programmfile**
Bereiche innerhalb eines Prozessors, die die Kontaktplanlogik- programme enthalten. MicroLogix-Steuerungen unterstützen mehrere Programmfiles.

**Programmiergerät**
Ein Programmierpaket zur Erstellung von Kontaktplandiagrammen.

**Programmmodus**
Wenn die das Steuerungsprogramm nicht durch die Steuerung abgefragt wird.

**Protokoll**
Die Regeln für den Datenaustausch über Kommunikationsver- bindungen.

**Prozessor**
Eine Systemeinheit (Central Processing Unit). (Siehe CPU.)

**Prozessorfile**
Die Programme und Datenfiles, die sich auf der Steuerung befinden.

**Pulsgeber**
Ein Gerät, das eine Position feststellt und ein Signal überträgt, das diese Position darstellt.

**Relais**
Ein elektrisch betriebenes Gerät, das elektrische Schaltungen mechanisch schaltet.

**Relaislogik**
Eine Darstellung binärer oder diskreter Logik.

**Remanente Daten**
Informationen (Daten), die beim Aus- und erneuten Einschalten der Spannungsversorgung nicht gelöscht werden.

**Reserviertes Bit**
Ein Speicherplatz, der für interne Zwecke reserviert ist.

**RS-232**
Ein EIA-Standard, in dem die elektrischen, mechanischen und funktionalen Merkmale serieller Binärkommunikationsschaltungen beschrieben werden.

**RTU**
Remote Terminal Unit

**Run-Modus**
Ein Ausführungsmodus, bei dem die Steuerung das Kontakt- planprogramm scannt oder ausführt.

**Schließkontakte**
Kontakte eines Relais oder Schalters, die geöffnet sind, wenn die Spannungsversorgung des Relais unterbrochen oder der Schalter deaktiviert wird. Die Kontakte werden geschlossen, wenn am Relais Spannung angelegt oder der Schalter aktiviert wird.

**Schreiben**
Übertragung von Daten an ein anderes Gerät. Beispielsweise verwen- det der Prozessor einen Nachrichten-Schreibbefehl, um Daten an ein anderes Gerät zu senden.

**Speichern**
Ein Programm auf der Festplatte eines Computers sichern.

**Sprung zur Marke**
Änderung der normalen Reihenfolge der Programmausführung. In Kontaktplanprogrammen verursacht ein Sprungbefehl (JMP) die Fort- führung des Programms in einem bestimmten Strompfad in dem Anwenderprogramm.

**Status**
Der Zustand einer Schaltung oder eines Systems.

**Steuerung**
Ein Gerät, z. B. eine speicherprogrammierbare Steuerung, zur Steuerung von Ausgangsgeräten.

**Steuerungs-Overhead**
Ein Teil des Arbeitszyklus, der für Verwaltungszwecke (Speicher- prüfung, Tests, Kommunikation usw.) verwendet wird.

**Steuerungsprofil**
Die Grundlage der Entscheidung der Steuerung, welche Ausgänge unter welchen Bedingungen eingeschaltet werden.

**Steuerungsprogramm**
Anwenderlogik (die Anwendung), die den Steuerungsbetrieb definiert.

**Stromliefernd**
Ein Begriff, mit dem der Stromfluss zwischen zwei Geräten beschrie- ben wird. Ein stromlieferndes Gerät oder eine stromliefernde Schaltung liefert Strom.

**Strompfad**
Ein Strompfad enthält Ein- und Ausgangsbefehle. Im Run-Modus werden die Eingänge in einem Strompfad als wahr oder unwahr erkannt. Wenn ein Suchpfad wahr ist, werden auch die Ausgänge auf wahr gesetzt (eingeschaltet). Wenn alle Pfade unwahr sind, werden auch die Ausgänge auf unwahr gesetzt (ausgeschaltet).

**Stromziehend**
Ein Begriff, mit dem der Stromfluss zwischen zwei Geräten beschrieben wird. Ein stromziehendes Gerät liefert eine direkte Erdungsverbindung.

**Unteres Byte**
Die Bits 0 bis 7 eines Worts.

**Unwahr**
Der Status eines Befehls, der in einem Kontaktplan-Strompfad keinen kontinuierlichen logischen Pfad darstellt.

**Verwaltung**
Der Teil der Abfrage, bei der die Steuerungen interne Prüfungen vornimmt und die Kommunikation bearbeitet.

**Vollduplex**
Ein Kommunikationsmodus, bei dem gleichzeitig Daten gesendet und empfangen werden (vgl. Halbduplex).

**Wahr**
Der Status eines Befehls, der in einem Kontaktplan-Strompfad einen kontinuierlichen logischen Pfad darstellt.

**Watchdog-Zeitwerk:**
Ein Zeitwerk, das einen zyklischen Prozess überwacht und am Ende eines jeden Zyklus gelöscht wird. Wenn der Watchdog die program- mierte Zeitspanne überschreitet, wird ein Fehler erzeugt.

**Wiederherstellen**
Übertragung eines Programms von einem Gerät zu einer Steuerung.

**Zähler**
Ein Gerät, das die Häufigkeit des Auftretens eines Ereignisses zählt.

**Zweig**
Ein logischer Parallelpfad in einem Strompfad eines Kontaktplan- programms. Haupteinsatzzweck eines Zweigs ist der Aufbau einer ODER-Logik.

## Ziffer

2-Kanal-Encoder *5-20*

## A

Abfallender Einzelimpuls *7-6*
Abfrage *G-1*
Abfragezeit *G-1*
ABL-Befehl *20-14*
ABS-Befehl *10-10*
Absolutwert *10-10*
Abwärtszählung *8-9*
ACB-Befehl *20-16*
ACI-Befehl *20-17*
ACL-Befehl *20-7*
ACN-Befehl *20-19*
ADD-Befehl *10-7*
Adresse *G-1*
Adressierung
  direkte Adressierung *4-3*
  E/A *1-9*
  eingeschleifte indirekte Adressierung verwenden *20-30*
  indirekte Adressierung *4-4*
  indirekte Adressierung eines Bits *4-6*
  indirekte Adressierung eines Files *4-5*
  indirekte Adressierung eines Worts *4-4*
  Modi *4-3*
  unmittelbare Adressierung *4-3*
AEX-Befehl *20-20*
AHL-Befehl *20-21*
AIC+ Erweiterter Schnittstellenwandler *G-1*
AIC-Befehl *20-8*
AND-Befehl *12-3*
Anfordern der Unterstützung von Rockwell Automation *D-11*
Anlauffehlerschutz, Bit *C-5*
Anwender-Interrupt aktivieren *18-10*
Anwender-Interrupt deaktivieren *18-9*
Anwender-Interrupt entfernen *18-11*
Anwendung *G-1*
Anzahl der ASCII-Zeichen im Puffer *20-16*
Arbeitsblatt zur Abfragezeit
  MicroLogix 1200 *A-7*
  MicroLogix 1500 *B-7*
ARD-Befehl *20-23*
Arithmetik-Flags *C-3*
ARL-Befehl *20-24*
ASC-Befehl *20-26*
ASCII
  Definition *G-1*
ASCII schreiben *20-12*
ASCII schreiben und anhängen *20-9*
ASCII-Befehle *20-1*
  Fehlercodes *20-31*
  Status-Bit *10-4*
  Status-Bits *20-5*, *20-6*, *22-14*
  Zeitdiagramm *20-29*
ASCII-File *20-5*
ASCII-Ganzzahl in Zeichenkette *20-8*
ASCII-Handshake-Leitungen *20-21*
ASCII-Lesen *20-23*
ASCII-Protokollparameter *20-4*
ASCII-Puffer auf Zeile überprüfen *20-14*
ASCII-Puffer löschen *20-7*
ASCII-Steuerdatenfile *20-6*
ASCII-Zeichenkette extrahieren *20-20*
ASCII-Zeichenkette in Ganzzahl *20-17*
ASCII-Zeichenkette suchen *20-26*
ASCII-Zeichenkette vergleichen *20-28*
ASCII-Zeichenkette verketten *20-19*
ASCII-Zeichensatz *20-32*
ASCII-Zeile lesen *20-24*
ASCII-Zeitdiagramm *20-29*
ASR-Befehl *20-28*
Auf geschlossen prüfen *7-1*
Auf offen prüfen *7-1*
Auffrischen *17-5*
Aufwärtszählung *8-9*
Ausführungsmodus *G-8*
Ausführungszeit
  MicroLogix 1200, Befehle *A-1*
  MicroLogix 1500, Befehle *B-1*
Ausgang entriegeln *7-4*
Ausgang verriegeln *7-4*
Ausgangsabfrage *G-1*
Ausgangsgerät *G-1*
Ausgehender Nachrichten-Befehl anstehend, Status-Bit *C-18*
Auswahl der mathematischen Überlauffunktion, Bit *C-11*
AWA- und AWT-Zeitdiagramm *20-29*
AWA-Befehl *20-9*
AWT-Befehl *20-12*

## B

Basis-Hardware-Information-File *3-13*
Batterie
  Betrieb *3-4*
  Lebensdauer *3-4*
Baudraten, Status *C-16*
BCD in Ganzzahl *11-4*
Befehl *G-2*
Befehlsausführungszeit *B-1*
Befehlssatz
  Definition *G-2*
  MicroLogix 1200, Ausführungszeiten *A-1*
  MicroLogix 1500, Ausführungszeiten *B-1*
  Übersicht *4-1*
Benutzeranwendungsmodusstatus *C-4*
Benutzerprogramm-Funktionstyp, Status *C-21*
Beseitigen
  Steuerungsfehler *D-1*
Bestellnummer, Status *C-20*, *C-21*
Betriebsspannung *G-2*

Betriebssytem
Bestellnummer, Status *C-20*
FRN, Status *C-21*
Serienkennung, Status *C-21*
BHI-Funktionsfile *3-13*
Bit *G-2*
Bit nach links verschieben *14-6*
Bit nach rechts verschieben *14-8*
Bitbefehle *7-1*
Bits für geringfügige Fehler *C-12*
Blockdiagramm *G-2*
Boolesche Operanden *G-2*
BSL-Befehl *14-6*
BSR-Befehl *14-8*
Byte-Tausch *14-19*
Byte-Tausch-Befehl *14-19*

## C

CLR-Befehl *10-9*
Compiler-Revision
Build-Nummer, Status *C-21*
Versionsnummer, Status *C-22*
COP-Befehl *14-4*
CPU (Central Processing Unit), Definition *G-2*
CPW-Befehl *14-2*
CS-Funktionsfile *3-14*
CTD-Befehl *8-9*
CTU-Befehl *8-9*

## D

DAT
Funktionsfile *3-10*
Konfiguration *3-10*
Datenfiles *2-2*, *2-7*
Ausgang (O) *2-7*
Beispiele für die E/A-Adressierung *1-21*
Bit (B) *2-7*
Datenfiles schützen *2-8*
Doppelwort (L) *2-7*
E/A-Bilder für Erweiterungs-Module (MicroLogix 1200) *1-4*
E/A-Bilder für Erweiterungsmodule (MicroLogix 1500) *1-11*
Eingang (I) *2-7*
Fließkomma (F) *2-7*, *10-4*
Ganzzahl (N) *2-7*
Nachrichtenfile (MG) *21-6*
Organisation und Adressierung *20-5*
PID (PD) *19-2*
programmierbarer Endschalter (PLS) *5-28*
Statusfile (S) *C-1*
Steuerung (R) *2-7*
ST-File (String) *20-5*
Zähler (C) *8-7*
Zeitwerk (T) *8-1*
Datenfile-Überschreibschutz inaktiv, Status-Bit *C-19*
Datenprotokollierung *22-7*
Datentafel *G-2*
DCD-Befehl *11-2*
Dekodierung 4 in 1 auf 16 *11-2*
Dezentrale Nachrichten *21-37*
DF1 Halbduplex-Protokoll *E-6*
DF1-Halbduplex-Protokoll
Beschreibung *E-6*
Konfigurationsparamter *E-7*
DF1-Vollduplex-Protokoll *E-5*
Beschreibung *E-5*
Konfigurationsparameter *E-5*
DH-485-Kommunikationsprotokoll *E-2*
Konfigurationsparameter *E-3*
DH-485-Netzwerk
Beschreibung *E-2*
Konfigurationsparameter *E-3*
Protokoll *E-2*
Token-Rotation *E-2*
DIN-Schiene *G-2*
DIV-Befehl *10-8*
Division *10-8*
DLG-Befehl *22-13*
DTE, Definition *G-2*
Durchsatz *G-3*

## E

E/A *G-3*
E/A auffrischen *17-5*
E/A-Adressierung *1-9*
E/A-Forcen *1-22*
E/A-Konfiguration *1-1*
Echtzeituhr
Anzeige-Bit für Batterieladezustand *3-4*
deaktivieren *3-3*
Funktionsfile *3-3*
Genauigkeit *3-4*
Echtzeituhr anpassen *3-5*
Eigentumsrechte-Zeitablauf *E-8*
EII-Funktionsfile *18-17*
Ein- und Ausgangsbefehle *17-1*
Ein-/Ausgangsstatusfile *3-19*
Eingangsabfrage *G-3*
Eingangsfilter *1-22*
Eingangsfilterwahl geändert, Status-Bit *C-14*
Eingangsgerät *G-3*
Eingeschleifte indirekte Adressierung *20-30*
Einschaltmodusverhalten, Bit *C-7*
Einschaltstrom *G-3*

Einstellpotentiometer *3-6*
Fehlerzustände
*3-6*
Funktionsfile *3-6*
Einzelimpuls *7-5*, *G-3*
EMI *G-3*
ENC-Befehl *11-3*
Encoder
2-Kanal-Encoder *5-20*
END-Befehl *16-5*
Energiesparfunktion (PST) *3-10*
EQU-Befehl *9-3*
Ereigniseingangs-Interrupts-Funktionsfile (EII) *18-17*
Erste Abfrage, Status-Bit *C-9*
Erweiterungs-E/A *1-3*, *1-9*
Analog-E/A-Konfiguration *1-5*, *1-13*
Diskrete E/A-Konfiguration *1-4*, *1-11*
Exklusives ODER *12-5*

## F

Fehlende Übereinstimmung bei Speichermodulkennwörtern, Status-Bit *C-13*
Fehler
automatische Beseitigung *D-1*
behebbare und nicht behebbare Fehler *18-6*
erkennen *D-1*
manuelle Fehlerbeseitigung mit der Fehlerroutine *D-2*
Fehler erkennen *D-1*
Fehlerbehandlungsroutine
Anwenderfehler-Unterprogramm erstellen *18-6*
behebbare und nicht behebbare Fehler *18-6*
Filenummern, Status *C-17*
Status-Bit für schwerwiegenden Fehler *C-12*
Fehlerbeseitigung *D-2*
Fehler-Bit für die ASCII-Zeichenkettenbearbeitung *C-14*
Fehlercode für schwerwiegende Fehler, Status *C-14*
Fehlercodes *D-1*, *D-3*
EII-Fehlercodes *18-18*
Fehlercode für schwerwiegende Fehler, Status *C-14*
Fehlercodes zu ASCII-Befehlen *20-31*
Fehlercodes zu MSG-Befehlen *21-43*
Fehlermeldungen und Fehlercodes *D-1*
Handbuch zur Fehlersuche *D-3*
HSC-Fehlercodes *5-5*
mathematische Status-Bits *10-3*
mathematische Überlauferkennung *10-3*
PID-Laufzeitfehler *19-17*
PTO-Fehlercodes *6-18*
PWM-Fehlercodes *6-26*
STI-Fehlercode *18-13*
Fehlercodes zu ASCII-Befehlen *20-31*
Fehlermeldungen *D-1*, *D-3*
Fehlerroutine
Anwendungsbeschreibung *18-6*
Betrieb während Hauptsteuerprogramm *18-2*
Filenummern, Status *C-17*
manuelle Fehlerbeseitigung *D-2*
Priorität bei Interrupts *18-4*
Fehlersuche *D-3*, *D-11*
automatische Fehlerbeseitigung *D-1*
Fehlerroutine verwenden *D-2*
manuelle Fehlerbeseitigung *D-2*
Steuerungsfehler erkennen *D-1*
Unterstützung durch Rockwell Automation *D-11*
Fehlerüberbrückung beim Einschalten, Bit *C-5*
Fernpaketunterstützung *E-4*
Festplatte *G-3*
FET *G-3*
FFL-Befehl *14-10*
FFU-Befehl *14-13*
FIFO (First-In-First-Out) *G-3*
FIFO entladen *14-13*
FIFO laden *14-10*
File *G-3*
File kopieren *14-4*
Filebefehle *14-1*
Filefüllung *14-5*
Filter, Eingänge *1-22*
FLL-Befehl *14-5*
Forcen aktiv, Status-Bit *C-4*
Forcen installiert, Status-Bit *C-5*
Forcen, Ein- und Ausgänge *1-22*
FRD
Befehl *11-4*
Beispiel
*11-6*
Freilaufender Takt *C-12*
Funktionsfile mit Speichermoduldaten *3-7*
bei Fehler laden *3-9*
Fehler überbrücken *3-8*
Funktionstyp *3-7*
immer laden *3-9*
Modul vorhanden *3-8*
Modusverhalten *3-9*
Programmvergleich *3-8*
Schreibschutz *3-8*
Funktionsfiles *3-1*, *3-2*
Basis-Hardware-Information (BHI) *3-13*
DAT-Funktionsfile *3-10*
Echtzeituhr (RTC) *3-3*
Ein-/Ausgangsstatusfile (IOS) *3-19*
Einstellpotentiometerdaten (TPI) *3-6*
Ereigniseingangs-Interrupts (EII) *18-17*
Hochgeschwindigkeitszähler (HSC) *5-2*
Impulsausgang (PTO) *6-6*
Kommunikations-Status-File (CS) *3-14*

Pulsweitenmodulation (PWM) *6-20*
Speichermoduldaten (MMI) *3-7*
wählbarer zeitgesteuerter Interrupt (STI) *18-12*

## G

GCD-Befehl *11-10*
Genauigkeit, Zeitwerk *8-3*
GEQ-Befehl *9-5*
Gleich *9-3*
Gray-Code *11-10*
Grenzwerttest *9-7*
Größer als *9-4*
Größer als oder gleich *9-5*
GRT-Befehl *9-4*

## H

Halbduplex *G-4*
Halt wegen schwerem Fehler, Status-Bit *C-9*
Handbücher, Verweise *2*
Hauptsteuerbefehl *16-5*
Hauptsteuerrelais (MCR) *G-4*
Herunterladen *G-4*
Hochgeschwindigkeitsausgänge *6-1*
Hochgeschwindigkeitszähler-Funktionsfile *5-2*
Hochladen *G-4*
HSC-Funktionsfile *5-2*
HSL-Befehl *5-26*

## I

IIM-Befehl *17-1*
Impulsgang
Befehl *6-1*
Funktionsfile *6-6*
Impulsspeicher-Eingänge *1-23*
In BCD *11-8*
In diesem Handbuch verwendete Konventionen *1*
INT-Befehl *18-7*
Integrierte E/A *1-1*
Interrupts
Anwender-Interrupt aktivieren (UIE) *18-10*
Anwender-Interrupt deaktivieren *18-9*
Anwender-Interrupt entfernen *18-11*
Fehlerbehandlungsroutine *18-6*
Interrupt-Befehle *18-7*
Interrupt-Unterprogramm (INT) *18-7*
Latenzzeit *18-5*
STI starten (STS) *18-8*
Übersicht *18-1*
Interrupt-Unterprogramm *18-7*
IOM-Befehl *17-4*
IOS-Funktionsfile *3-19*
Istwert zurücksetzen, Befehl *5-27*

## J

JMP-Befehl *16-1*
JSR-Befehl *16-2*

## K

Kanal 0
CS0-Kommunikations-Status-File *3-14*
Kommunikationsstatus *C-17*
Kanalkonfiguration
DF1-Halbduplex-Parameter *E-7*
DF1-Vollduplex-Parameter *E-5*
DH-485-Parameter *E-3*
Modbus-RTU-Slave-Parameter *E-14*
Kennwortschutz *2-11*
Kleiner als *9-4*
Kleiner als oder gleich *9-5*
Klemme *G-4*
Kodierung 1 auf 16 in 4 *11-3*
Kommunikation
Aktiv, Status-Bit *C-18*
Kanal 0, Status *C-17*
Modusauswahl, Status-Bit *C-18*
Statusfile *3-14*
Kommunikationsabfrage *G-4*
Kommunikationsbearbeitung *21-3*
Kommunikationsbefehle *21-1*
Kommunikationsprotokoll
DF1-Halbduplex *E-6*
Kommunikationsprotokolle
DF1-Vollduplex *E-5*
DH-485 *E-2*
Modbus-Slave-RTU *E-9*
Kontaktplanlogik *G-4*
Konvertierung *11-1*

## L

Ladebefehl für Hochgeschwindigkeitszähler *5-26*
Ladestatus Batterie Status-Bit *C-14*
LBL-Befehl *16-2*
Leckstrom im AUS-Zustand *G-4*
LED (Leuchtdiode) *G-4*
LEQ-Befehl *9-5*
LES-Befehl *9-4*
Lesen *G-4*
Letzte 100 µs Scanzeit, Status *C-18*
LFL-Befehl *14-15*
LFU-Befehl *14-17*
LIFO (Last-In-First-Out) *G-5*
LIFO entladen *14-17*
LIFO laden *14-15*
LIM-Befehl *9-7*
Literaturhinweise *2*

Logik *G-5*
Logikbefehle *12-1*
Logisches NICHT *12-6*
Logisches ODER *12-4*
Logisches UND *12-3*
Löschen *10-9*
  Speicher der Steuerung *2-12*

## M

Marke *16-2*
Maskierte Verschiebung *13-3*
Maskierter Vergleich auf gleich *9-6*
Mathematische Befehle *10-1*
MCR-Befehl *16-5*
MEQ *9-6*
MEQ-Befehl *9-6*
MMI-Funktionsfile *3-7*
Mnemonik *G-5*
Modbus-Definition *G-5*
Modbus-Slave-RTU-Protokoll *E-9*
Modbus-zu-MicroLogix-Speicherbelegung *E-11*, *E-12*, *E-13*
Modem *G-5*
Modi *G-5*
Modusstatus *C-4*
Modusverhalten *C-8*
MOV-Befehl *13-1*
MSG-Befehl *21-5*
  Beispiele für zentrale Nachrichten *21-25*
  Fehlercodes *21-43*
  Kontaktplanlogik *21-15*
  Zeitdiagramm *21-12*
MUL-Befehl *10-8*
Multiplikation *10-8*
MVM-Befehl *13-3*

## N

Nachricht *21-5*
Nachrichten
  Beispiele für zentrale Nachrichten *21-25*
  dezentral *21-37*
  zentral *21-16*
Nachrichtenantwort anstehend, Status-Bit *C-17*
Nachrichtenfehler *21-43*
Nachrichtenfile (MG) *21-6*
Nachrichtenfunktion *21-1*
Negation *10-9*
Negative Logik *G-5*
NEG-Befehl *10-9*
NEQ-Befehl *9-3*
Netzknotenadressen, Status *C-16*
Netzwerk *G-5*
Niederwertigstes Bit (LSB) *G-5*
Nominaler Eingangsstrom *G-5*
NOT-Befehl *12-6*
Null-Flag *C-3*

## O

Oberes Byte *G-5*
OEM-Sperre *2-13*
OEM-Sperre, Status-Bit *C-9*
Offline *G-5*
Öffnungskontakte *G-5*
Offset *G-6*
Online *G-6*
ONS-Befehl *7-5*
OR-Befehl *12-4*
OSF-Befehl *7-6*
OSR-Befehl *7-6*
OTE-Befehl *7-3*
OTL-Befehl *7-4*
OTU-Befehl *7-4*

## P

PCCC *G-6*
PD-Datenfile *19-2*
PID
  Abstimmparameter *19-8*
  Analog-E/A-Skalierung *19-18*
  Anwendungsbeispiele *19-23*
  Anwendungshinweise *19-19*
  Fehler *19-17*
  PID- Gleichung *19-2*
  PID-Befehl *19-3*
  PID-Konzept *19-1*
PLS-File *5-28*
Programmabfrage
  Arbeitsblatt zur MicroLogix 1200-Abfragezeit *A-7*
  Arbeitsblatt zur MicroLogix 1500-Abfragezeit *B-7*
  Definition *G-6*
Programmende *16-5*
Programmfile
  Definition *G-6*
Programmierbarer Endschalter-File *5-28*
Programmiergerät *G-6*
Programmmodus *G-6*
Programmsteuerungsbefehle *16-1*
Proportional-/Integral-/Differenzialverhalten
  Anwendungshinweise *19-19*
  Laufzeitfehler *19-17*
  PID-Befehl *19-3*
  PID-Gleichung *19-2*
  PID-Konzept *19-1*
  PID-Regelkreis einstellen *19-23*

Protokoll *G-6*
DF1-Halbduplex *E-6*
DF1-Vollduplex *E-5*
DH-485-Kommunikation *E-2*
Modbus-Slave-RTU *E-9*
Protokollkonfiguration *E-1*
Prozessor *G-6*
Prozessor-Bestellnummer, Status *C-21*
Prozessorfile *G-6*
Prozessorserie, Status *C-21*
Prozessorversion, Status *C-21*
Prozesssteuerungsbefehl *19-1*
PTO
Befehl *6-1*
Funktionsfile *6-6*
Publikationen, Verweise *2*
Pulsdauermodulation
Befehl *6-19*
Pulsgeber
Definition *G-6*
Pulsweitenmodulation
Funktionsfile *6-20*
PWM
Befehl *6-19*
Funktionsfile *6-20*

## Q

Quadratwurzel *10-15*

## R

RAC-Befehl *5-27*
RCP-Befehl *22-1*
Rechenregister, Status *C-16*
REF-Befehl *17-5*
Relais *G-6*
Relaisbefehle *7-1*
Relaislogik *G-6*
Remanente Daten *G-6*
Remanente Daten verloren, Status-Bit *C-13*
RES-Befehl *8-10*
Reserviertes Bit *G-7*
RET-Befehl *16-3*
Rezept *22-1*
Rockwell Automation
Unterstützung anfordern *D-11*
RS-232, Definition *G-7*
RTA-Befehl *3-5*
RTC
Funktionsfile *3-3*
Jahr, Status *C-19*
Minuten, Status *C-20*
Monat, Status *C-19*
Sekunden, Status *C-20*
Stunden, Status *C-20*
Tag, Status *C-19*
Wochentag, Status *C-20*
RTO-Befehl *8-6*
RTU, Definition *G-7*
Rückkehr vom Unterprogramm *16-3*
Run-Modus *G-7*

## S

SBR-Befehl *16-3*
Scanzeit
Letzte 100 µs Scanzeit, Status *C-18*
Status für maximale Scanzeit *C-16*
Schließkontakte *G-7*
Schreiben *G-7*
Schrittschaltwerksausgang *15-6*
Schrittschaltwerksbefehle *15-1*
Schrittschaltwerksladung *15-9*
Schrittschaltwerksvergleich *15-2*
Schwerwiegender Fehler in Benutzerfehlerroutine, Status-Bit *C-12*
SCL-Befehl *10-12*
SCP-Befehl *10-13*
Skalierung *10-12*
Skalierung mit Parametern *10-13*
Sofortiger Ausgang mit Maske *17-4*
Sofortiger Eingang mit Maske *17-1*
Speicher *2-2*
Speicher der Steuerung löschen *2-12*
Speicherbelegung
MicroLogix 1200, Befehle *A-1*
MicroLogix 1500, Befehle *B-1*
Steuerungsspeicherbelegung überprüfen *2-6*
Speichermodul bei Fehler oder Standardprogramm laden, Bit *C-6*
Speichermodul immer laden, Bit *C-6*
Speichermodul laden, Status-Bit *C-13*
Speichermodulvergleich, Bit *C-10*
Speichern *G-7*
Speichernder Timer *8-6*
Sprung ins Unterprogramm *16-2*
Sprung zu Marke *16-1*
Sprung zur Marke *G-7*
SQC-Befehl *15-2*
SQL-Befehl *15-9*
SQO-Befehl *15-6*
SQR-Befehl *10-15*
Standard
Ausgangsanordnung *1-18*
Statischer Fileschutz *2-10*
Status *G-7*
Status freilaufender Takt *C-12*

Status für aktive Netzknoten *C-15*
Status für maximale Scanzeit *C-16*
Status-Bit für schwache Batterie *C-14*
Statusfile *C-1*
Steigender Einzelimpuls *7-6*
Steuerregisterfehler, Status-Bit *C-12*
Steuerung
  Definition *G-8*
  Fehlermeldungen *D-3*
  Modus *C-8*
  Modusstatus *C-4*
  Overhead *A-7*, *B-7*, *G-7*
  Statusfile *C-1*
Steuerungsbetrieb überwachen, Fehlerbeseitigung *D-2*
Steuerungsfehler erkennen *D-1*
Steuerungsprofil *G-7*
Steuerungsprogramm *G-7*
STI
  Aktiviert, Bit *C-10*
  Anstehend, Status-Bit *C-10*
  Ausführungs-Bit *C-10*
  Filenummern, Status *C-17*
  Funktionsfile *18-12*
  Modusstatus *C-10*
  Sollwert, Status *C-17*
  Verlust, Status-Bit *C-13*
STI starten *18-8*
STI-Funktionsfile *18-12*
Stromliefernd *G-8*
Strompfad *G-8*
Stromziehend *G-8*
STS-Befehl *18-8*
SUB-Befehl *10-7*
Subtraktion *10-7*
SUS-Befehl *16-4*
Suspend-Code, Status *C-15*
Suspend-File, Status *C-15*
Suspend-Zustand *16-4*
SVC-Befehl *21-3*
SWP-Befehl *14-19*

## T

Takt, freilaufend *C-12*
Temporäres Ende *16-4*
Timer-Ausschaltverzögerung *8-5*
Timer-Einschaltverzögerung *8-4*
TND-Befehl *16-4*
TOD-Befehl *11-8*
  Änderungen im Rechenregister *11-9*
  Beispiel
    *11-9*
TOF-Befehl *8-5*
TON-Befehl *8-4*
TPI-Funktionsfile *3-6*
Typische Netzwerkkonfiguration *21-29*

## U

Überlauferkennung, Status-Bit *C-12*
Überlauf-Flag *C-3*
Übertrag-Flag *C-3*
Übertragungsgeschwindigkeit *G-2*
UID-Befehl *18-9*
UIE-Befehl *18-10*
UIF-Befehl *18-11*
Umschaltabfrage, Status-Bit *C-18*
Ungleich *9-3*
Unteres Byte *G-8*
Unterprogramm *16-3*
Unwahr *G-8*

## V

Vergleichsbefehle *9-1*
Verschiebebefehle *13-1*
Verwaltung *G-8*
Vollduplex *G-8*

## W

Wahr *G-8*
Watchdog-Abfragezeit *C-11*
Wiederherstellen *G-8*
Wort kopieren *14-2*

## X

XIC-Befehl *7-1*
XIO-Befehl *7-1*
XOR-Befehl *12-5*

## Z

Zähler
  Definition *G-8*
  Funktionsweise *8-7*
  Steuerfile- und Status-Bits *8-8*
  Zählerfile *8-7*
Zeichen-Flag *C-4*
Zeichenketten-Datenfile *20-5*
Zeitdiagramm
  Impulsspeicher-Eingänge *1-23*
Zeitdiagramme
  2-Kanal-Encoder *5-20*
  ASCII *20-29*
  AWA- und AWT-Befehle *20-29*
  MSG-Befehl *21-12*
  relative PTO-Zeitregelung *6-4*
Zeitwerk- und Zählerbefehle *8-1*

Zeitwerkfiles *8-1*
Zeitwerkgenauigkeit *8-3*
Zentrale Nachrichten *21-16*
Ziel-Bit-File *3-10, 3-12*
Ziel-Ganzzahl-File *3-10*
Zukünftigen Zugriff zulassen *2-13*
Zukünftiger Zugriff, Status-Bit *C-9*
Zurücksetzen *8-10*
Zweck dieses Handbuchs *1*
Zweig *G-8*

# MicroLogix 1200 und 1500
# Alphabetische Liste der Befehle


| Befehlsbeschreibung | Seite |
|---|---|
| ABL – Puffer auf Zeile überprüfen | 20-14 |
| ABS – Absolutwert | 10-10 |
| ACB – Anzahl der ASCII-Zeichen im Puffer | 20-16 |
| ACI – Zeichenkette in Ganzzahl | 20-17 |
| ACL – ASCII-Puffer löschen | 20-7 |
| ACN – Zeichenkette verketten | 20-19 |
| ADD – Addition | 10-7 |
| AEX – Zeichenkette extrahieren | 20-20 |
| AHL – ASCII-Handshake- Leitungen | 20-21 |
| AIC – ASCII-Ganzzahl in Zeichenkette | 20-8 |
| AND – Logisches UND | 12-3 |
| ARD – ASCII-Lesen | 20-23 |
| ARL – ASCII-Zeile lesen | 20-24 |
| ASC – Zeichenkette suchen | 20-26 |
| ASR – ASCII-Zeichenkette vergleichen | 20-27 |
| AWA – ASCII schreiben und anhängen | 20-9 |
| AWT – ASCII schreiben | 20-12 |
| BSL – Bit nach links verschieben | 14-6 |
| BSR – Bit nach rechts verschieben | 14-8 |
| CLR – Löschen | 10-9 |
| COP – File kopieren | 14-4 |
| CPW – Wort kopieren | 14-2 |
| CTD – Abwärtszählung | 8-9 |
| CTU – Aufwärtszählung | 8-9 |
| DCD – 4 in 1 auf 16 dekodieren | 11-2 |
| DIV – Division | 10-8 |
| DLG – Datenprotokollbefehl | 22-13 |
| ENC – Kodierung 1 auf 16 in 4 | 11-3 |
| END – Programmende | 16-5 |
| EQU – Gleich | 9-3 |
| FFL – FIFO laden | 14-10 |
| FFU – FIFO entladen | 14-13 |
| FLL – Filefüllung | 14-5 |
| FRD – BCD in Ganzzahl | 11-4 |
| GCD – Gray-Code | 11-10 |
| GEQ – Größer als oder gleich | 9-5 |
| GRT – Größer als | 9-4 |
| HSL – Hochgeschwindigkeitszähler laden | 5-26 |
| IIM – Sofortiger Eingang mit Maske | 17-1 |
| INT – Interrupt- Unterprogramm | 18-7 |
| IOM – Sofortiger Ausgang mit Maske | 17-4 |
| JMP – Sprung zu Marke | 16-1 |
| JSR – Sprung ins Unterprogramm | 16-2 |
| LBL – Marke | 16-2 |
| LEQ – Kleiner als oder gleich | 9-5 |
| LES – Kleiner als | 9-4 |
| LFL – LIFO laden | 14-15 |
| LFU – LIFO entladen | 14-17 |

| Befehlsbeschreibung | Seite |
|---|---|
| LIM – Grenzwerttest | 9-7 |
| MCR – Hauptsteuerbefehl | 16-5 |
| MEQ – Maskierter Vergleich auf gleich | 9-6 |
| MOV – Verschieben | 13-1 |
| MSG – Nachricht | 21-5 |
| MUL – Multiplikation | 10-8 |
| MVM – Maskierte Verschiebung | 13-3 |
| NEG – Negation | 10-9 |
| NEQ – Ungleich | 9-3 |
| NOT – Logisches NICHT | 12-6 |
| ONS – Einzelimpuls | 7-5 |
| OR – Logisches ODER | 12-4 |
| OSF – Fallender Einzelimpuls | 7-6 |
| OSR – Steigender Einzelimpuls | 7-6 |
| OTE – Ausgang einschalten | 7-3 |
| OTL – Ausgang setzen | 7-4 |
| OTU – Ausgang rücksetzen | 7-4 |
| PID – Proportional-/Integral-/Differenzialverhalten | 19-3 |
| PTO – Pulse Train Output (Impulsausgang) | 6-1 |
| PWM – Pulsweitenmodulation | 6-19 |
| RAC – Istwert zurücksetzen | 5-27 |
| RCP – Rezept (nur MicroLogix 1500) | 22-1 |
| REF – E/A auffrischen | 17-5 |
| RES – Zurücksetzen | 8-10 |
| RET – Rückkehr vom Unterprogramm | 16-3 |
| RTA-Befehl (Echtzeituhr anpassen) | 3-5 |
| RTO – Speichernder Timer Ein | 8-6 |
| SBR – Unterprogramm | 16-3 |
| SCL – Skalierung | 10-12 |
| SCP – Skalierung mit Parametern | 10-13 |
| SQC – Schrittschaltwerksvergleich | 15-2 |
| SQL – Schrittschaltwerksladung | 15-8 |
| SQO – Schrittschaltwerksausgang | 15-5 |
| SQR – Quadratwurzel | 10-15 |
| STS – STI starten | 18-8 |
| SUB – Subtraktion | 10-7 |
| SUS – Suspend | 16-4 |
| SWP – Byte-Tausch | 14-19 |
| TND – Temporäres Ende | 16-4 |
| TOD – In BCD | 11-8 |
| TOF – Timer-Ausschaltverzögerung | 8-5 |
| TON – Timer-Einschaltverzögerung | 8-4 |
| UID – Anwender-Interrupt deaktivieren | 18-9 |
| UIE – Anwender-Interrupt aktivieren | 18-10 |
| UIF – Anwender-Interrupt entfernen | 18-11 |
| XIC – Auf geschlossen prüfen | 7-1 |
| XIO – Auf offen prüfen | 7-1 |
| XOR – Exklusives ODER | 12-5 |

**www.rockwellautomation.com**

**Weltweite Hauptverwaltung**
Rockwell Automation, 777 East Wisconsin Avenue, Suite 1400, Milwaukee, WI 53202-5302, USA, Tel.: +1 414 212 52 00, Fax: +1 414 212 52 01

**Hauptverwaltung für Allen-Bradley, Rockwell Software und Global Manufacturing Solutions**
Europa/Naher Osten/Afrika: Rockwell Automation SA/NV, Vorstlaan/Boulevard du Souverain 36, 1170 Brüssel, Belgien, Tel.: +32 (0)2 663 06 00, Fax: +32 (0)2 663 06 40

**Hauptverwaltung für Dodge und Reliance Electric**
Europa/Naher Osten/Afrika: Rockwell Automation, Brühlstraße 22, 74834 Elztal-Dallau, Deutschland, Tel.: +49 (0)6261 9410, Fax: +49 (0)6261 17741

**Hauptverwaltung Deutschland,** Düsselberger Straße 15, 42781 Haan, Tel.: +49 (0)2104 960 0, Fax: +49 (0)2104 960 121, www.rockwellautomation.de
**Verkaufs- und Supportzentrum Schweiz,** Hintermättlistraße 3, 5506 Mägenwil, Tel.: +41 (0)62 889 77 77, Fax: +41 (0)62 889 77 66, www.rockwellautomation.ch
**Hauptverwaltung Österreich,** Kotzinastraße 9, 4030 Linz, Tel.: +43 (0)732 38 909 0, Fax: +43 (0)732 38 909 61, www.rockwellautomation.at

Publikation 1762-RM001D-DE-P – Oktober 2002 PN 40072-079-03(4)
Ersetzt Publikation 1762-RM001C-DE-P – November 2000
© 2002 Rockwell Automation. Alle Rechte vorbehalten. Printed in USA.